

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ МІСЬКОГО
ГОСПОДАРСТВА імені О. М. БЕКЕТОВА
Навчально-науковий Інститут енергетичної, інформаційної та
транспортної інфраструктури
Кафедра автоматизації та комп'ютерно-інтегрованих технологій

РОЗРАХУНКОВО-ПОЯСНЮВАЛЬНА ЗАПИСКА
ДО КВАЛІФІКАЦІЙНОЇ РОБОТИ БАКАЛАВРА

на тему: «Система моніторингу та аналізу енергоспоживання електромобіля
на основі експлуатаційних даних»

Виконав: здобувач вищої освіти
4 курсу, групи Сінж 2022-1
напряму підготовки (спеціальності)
151 «Автоматизація та комп'ютерно-
інтегровані технології»
Репкін Олег Радієвич

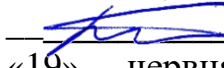
Керівник: Арсеньєва О. П., д.т.н., проф.
Рецензент: Капустенко П.О., к.т.н., доц.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
МІСЬКОГО ГОСПОДАРСТВА імені О. М. БЕКЕТОВА

**Навчально-науковий Інститут енергетичної, інформаційної та
транспортної інфраструктури**
Кафедра автоматизації та комп'ютерно-інтегрованих технологій
Освітньо-кваліфікаційний рівень – бакалавр
Галузі знань 17 «Електроніка, автоматизація та електронні комунікації»
Спеціальність 151 «Автоматизація та комп'ютерно-інтегровані технології»

ЗАТВЕРДЖУЮ

В.о. завідувача кафедри АКІТ

 **БАРАНОВ О. О.**
«19» червня 2026 року

З А В Д А Н Н Я
НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА

Репкін Олег Радієвич

(прізвище, ім'я, по батькові)

1. Тема роботи «Система моніторингу та аналізу енергоспоживання електромобіля на основі експлуатаційних даних»

затвержені наказом закладу вищої освіти від «22» травня 2026 р. № 440-03

керівник роботи Арсеньєва О. П., д.т.н., проф.

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

2. Термін подання студентом роботи 15.06.2026 р.

3. Вихідні дані до роботи: рекомендації щодо розробки додатку, індивідуальне завдання на розробку, дані діагностики електромобіля









4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

Аналіз предметної області: дослідити особливості енергоспоживання електромобіля та фактори, що впливають на витрати електроенергії. Розглянути методи моніторингу експлуатаційних даних із використанням OBD, CAN та BMS. Розробити структуру програмної системи моніторингу та аналізу енергоспоживання. Визначити склад вхідних даних та алгоритм їх завантаження. Реалізувати програмне забезпечення для обробки експлуатаційних даних, розрахунку енергетичних показників та аналізу стану акумуляторної батареї. Провести тестування системи, аналіз результатів та виконати розрахунки з охорони праці.


5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

Технологічна схема реалізації програмної системи аналізу енергоспоживання. Блок-схема архітектури програмної системи розшифрування та аналізу експлуатаційних даних електромобіля. Графіки аналізу швидкості, потужності, рівня заряду, температури батареї та енергоспоживання.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Розділ I	О. П. Арсеньєва, професор		
Розділ II	О. П. Арсеньєва, професор		
Розділ III	О. П. Арсеньєва, професор		
Розділ IV	В. В. Малишева, доцент		

7. Дата видачі завдання 11.05.2026 р.

Керівник 
(підпис)

Завдання прийняв до виконання _____
(підпис)

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Предпроектні дослідження	12.05.2026	
2	Аналіз предметної області	14.05.2026	
3	Розробка алгоритму планування підзарядки батареї	16. 05.2026	
4	Розробка програмного забезпечення	24.05.2026	
5	Тестування і аналіз результатів роботи програмного продукту	06.05.2026	
6	Охорона праці в галузі	08. 06.2026	

Студент



(підпис)

О. Р. Репкін

(прізвище та ініціали)

Керівник роботи



(підпис)

О. П. Арсеньєва

(прізвище та ініціали)

РЕФЕРАТ

У роботі представлено розроблення програмної підсистеми, призначеної для дослідження впливу швидкості руху на споживання енергії електромобілем. Об'єктом дослідження є процес витрачання електричної енергії електромобілем у різних режимах руху. Предмет дослідження охоплює способи опрацювання даних, отриманих через інтерфейс OBD, математичне моделювання взаємозв'язку між швидкістю та питомими витратами енергії, а також програмні інструменти для аналізу й наочного подання отриманих результатів. Метою роботи є створення програмної підсистеми для визначення характеру впливу швидкісного режиму на енергоспоживання електромобіля, вибору найбільш енергоефективної швидкості руху та розрахунку прогнозованого запасу ходу.

Структура роботи представлена вступом, чотирма розділами, висновками, переліком посилань.

Структура та обсяг дипломної роботи бакалавра. Випускна кваліфікаційна робота бакалавра складається зі вступу, 4-х розділів, висновків, додатків та переліку посилань. Повний обсяг випускної кваліфікаційної роботи бакалавра становить 95 сторінок, в тому числі: 10 рисунків; 2 таблиці; 1 додаток на 16 сторінках; перелік посилань з 43 найменувань на 5 сторінках.

КЛЮЧОВІ СЛОВА: ЕЛЕКТРОМОБІЛЬ, ЕНЕРГОСПОЖИВАННЯ, ШВИДКІСНИЙ РЕЖИМ, ЗАПАС ХОДУ, РЕКУПЕРАЦІЯ.

ABSTRACT

The study presents the development of a software subsystem intended to investigate the effect of vehicle speed on electric vehicle energy consumption. The object of the research is the process of electrical energy use by an electric vehicle under different driving conditions. The subject of the research includes methods for processing data obtained through the OBD interface, mathematical modelling of the relationship between vehicle speed and specific energy consumption, as well as software tools for analysing and visualising the obtained results. The aim of the study is to develop a software subsystem for assessing the influence of driving speed on electric vehicle energy consumption, identifying the most energy-efficient speed range, and estimating the expected driving range.

The structure of the work is represented by an introduction, four sections, conclusions and a list of references.

Structure and scope of the bachelor's thesis. The bachelor's thesis consists of an introduction, 4 sections, conclusions, appendices and a list of references. The full volume of the bachelor's final qualification work is 95 pages, including: 10 drawings; 2 table; 1 supplement on 16 pages; a list of links with 43 titles on 5 pages.

KEYWORDS: ELECTRIC VEHICLE, ENERGY CONSUMPTION, SPEED MODE, DRIVING RANGE, REGENERATIVE BRAKING.

ЗМІСТ

ВСТУП.....	8
РОЗДІЛ 1. ТЕОРЕТИЧНІ ОСНОВИ МОНІТОРИНГУ ЕНЕРГОСПОЖИВАННЯ ЕЛЕКТРОМОБІЛЯ.....	11
1.1. Аналіз особливостей енергоспоживання електромобілів.....	11
1.2. Методи моніторингу та аналізу експлуатаційних даних	16
1.3. Обґрунтування структури системи моніторингу	20
Висновки до першого розділу.....	25
РОЗДІЛ 2. РОЗРОБКА СИСТЕМИ МОНІТОРИНГУ ТА АНАЛІЗУ ЕНЕРГОСПОЖИВАННЯ ЕЛЕКТРОМОБІЛЯ	27
2.1. Проектування структури програмної системи.....	27
2.2. Розробка програмного забезпечення системи.....	36
2.3. Реалізація функцій моніторингу енергоспоживання.....	41
Висновки до другого розділу	44
РОЗДІЛ 3. ДОСЛІДЖЕННЯ ЕФЕКТИВНОСТІ РОЗРОБЛЕНОЇ СИСТЕМИ....	46
3.1. Тестування програмної системи.....	46
3.2. Аналіз результатів моніторингу енергоспоживання	53
3.3. Перспективи вдосконалення системи	55
Висновки до третього розділу	58
РОЗДІЛ 4. ОХОРОНА ПРАЦІ	60
4.1. Організаційно-правові основи забезпечення безпеки праці.....	60
4.2. Характеристика об'єкта та виявлення потенційних небезпек.....	63
4.3. Дослідження ризику реалізації небезпек та заходи безпеки	67
Висновки до четвертого розділу.....	70

ВИСНОВКИ.....	72
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	75
Додаток А.....	80

ВСТУП

Стрімкий розвиток електромобільного транспорту є однією з ключових тенденцій сучасної транспортної галузі. Зростання вимог до енергоефективності, зменшення викидів парникових газів та скорочення використання викопних видів палива сприяють активному впровадженню електромобілів у різних країнах світу. За останні роки значно збільшилася кількість електротранспортних засобів, удосконалилися технології виробництва тягових акумуляторів та систем керування електроприводом, а також розширилася інфраструктура зарядних станцій.

Електромобілі характеризуються високим коефіцієнтом корисної дії електроприводу, низькими експлуатаційними витратами та можливістю використання відновлюваних джерел енергії. Разом із тим ефективність їх експлуатації значною мірою залежить від режимів руху, технічного стану акумуляторної батареї, умов навколишнього середовища та особливостей керування транспортним засобом. У процесі експлуатації електромобіль формує значний обсяг телеметричних даних, які містять інформацію про напругу батареї, струм, температуру, швидкість руху, рівень заряду та інші параметри роботи транспортного засобу.

Сучасні системи діагностики та моніторингу дозволяють отримувати великий обсяг експлуатаційної інформації через OBD-інтерфейси, CAN-шину та систему керування батареєю BMS. Однак для ефективного використання таких даних необхідне створення спеціалізованого програмного забезпечення, здатного автоматично виконувати їх обробку, аналіз та візуалізацію результатів. Використання подібних програмних засобів дозволяє оцінювати енергетичну ефективність транспортного засобу, контролювати технічний стан батареї та виявляти аномальні режими роботи.

Особливого значення набуває аналіз енергоспоживання електромобілів на основі реальних експлуатаційних даних. На відміну від лабораторних випробувань, аналіз реальних даних дозволяє враховувати вплив дорожніх умов,

стилю керування, температурного режиму та інших факторів, що безпосередньо впливають на витрати електроенергії та ресурс акумуляторної батареї.

Актуальність теми дослідження зумовлена необхідністю підвищення енергоефективності електромобілів та створення програмних засобів автоматизованого аналізу експлуатаційних даних. Застосування таких систем дозволяє здійснювати моніторинг стану транспортного засобу, визначати закономірності споживання електроенергії, виявляти потенційні несправності та підвищувати ефективність використання акумуляторної батареї.

Метою роботи є розробка програмної системи моніторингу та аналізу енергоспоживання електромобіля на основі експлуатаційних даних, яка забезпечує автоматичне зчитування, обробку, аналіз та візуалізацію параметрів роботи транспортного засобу.

Для досягнення поставленої мети необхідно вирішити такі **завдання**:

- провести аналіз особливостей енергоспоживання електромобілів та факторів, що впливають на витрати електроенергії;
- дослідити сучасні методи збору та аналізу телеметричних даних транспортних засобів;
- обґрунтувати структуру програмної системи моніторингу;
- розробити програмне забезпечення для автоматичного зчитування та обробки експлуатаційних даних;
- реалізувати алгоритми аналізу енергоспоживання та оцінювання стану акумуляторної батареї;
- виконати тестування програмної системи на реальних експлуатаційних даних;
- провести аналіз отриманих результатів та визначити перспективи подальшого розвитку системи.

Об'єкт дослідження: процеси енергоспоживання електромобіля під час його експлуатації.

Предмет дослідження: методи та програмні засоби моніторингу, обробки та аналізу експлуатаційних даних електромобіля.

Методи дослідження базуються на використанні методів аналізу часових рядів, математичної обробки експлуатаційних даних, статистичного аналізу, програмування мовою Python та методів візуалізації інформації.

Практичне значення одержаних результатів полягає у розробці програмної системи, яка дозволяє автоматично обробляти телеметричні дані електромобіля, розраховувати показники енергоспоживання, аналізувати стан акумуляторної батареї, виявляти аномальні режими роботи та формувати аналітичні звіти. Розроблене програмне забезпечення може бути використане для моніторингу електромобілів, проведення діагностичних досліджень та подальшого розвитку інтелектуальних транспортних систем.

Структура роботи: робота складається зі вступу, трьох розділів, висновків та списку використаних джерел. У першому розділі виконано аналіз особливостей енергоспоживання електромобілів та методів обробки експлуатаційних даних. Другий розділ присвячений розробці програмної системи моніторингу та аналізу енергоспоживання. В третьому розділі наведено результати тестування програмного забезпечення, аналіз отриманих даних та перспективи подальшого розвитку системи.

РОЗДІЛ 1. ТЕОРЕТИЧНІ ОСНОВИ МОНІТОРИНГУ ЕНЕРГОСПОЖИВАННЯ ЕЛЕКТРОМОБІЛЯ

1.1. Аналіз особливостей енергоспоживання електромобілів

У процесі експлуатації сучасних електромобілів одним із ключових параметрів ефективності транспортного засобу є рівень споживання електричної енергії під час руху. На відміну від автомобілів із двигунами внутрішнього згоряння, де основна увага приділяється витратам пального, в електромобілях критично важливим показником виступає кількість електроенергії, яка використовується силовою установкою, системами керування, допоміжним обладнанням та кліматичними модулями. Практичний аналіз експлуатаційних характеристик показує, що фактичне енергоспоживання електромобіля може суттєво відрізнитися навіть при однакових маршрутах руху, бо на цей показник впливає велика кількість динамічних факторів, які змінюються в режимі реального часу.

Під час дослідження особливостей енергоспоживання електромобілів основна увага приділяється аналізу параметрів тягової батареї, роботи електродвигуна, режимів прискорення, швидкісних характеристик, рекуперативного гальмування та навантаження на силову електроніку. На практиці система керування електромобілем постійно виконує балансування між необхідною потужністю, температурним режимом батареї та допустимими струмовими навантаженнями. Через це, навіть, короткочасні зміни режиму руху можуть спричинити різке зростання споживання енергії.

Технічний аналіз показує, що найбільше навантаження на акумуляторну батарею виникає у моменти інтенсивного розгону транспортного засобу. В цей момент контролер електроприводу формує підвищений струм живлення електродвигуна, внаслідок чого миттєве енергоспоживання може в декілька разів перевищувати середні значення. Наприклад, при рівномірному русі міським маршрутом електромобіль може споживати близько 12–16 кВт·год на 100 км,

тоді як під час агресивного стилю керування цей показник збільшується до 22–28 кВт·год на 100 км. Особливо помітним це стає при різких прискореннях, частих обгонах та експлуатації автомобіля в режимі високих динамічних навантажень.

Важливим фактором, також, є маса транспортного засобу. Зі збільшенням ваги електромобіля підвищується навантаження на тяговий електродвигун, що безпосередньо впливає на рівень струму, який споживається від акумуляторної батареї. Додаткове навантаження у вигляді пасажирів, вантажу, або причепа призводить до збільшення енергоспоживання, особливо під час руху в умовах підйомів або інтенсивного міського циклу. В практичних вимірюваннях збільшення маси автомобіля на 100–150 кг може спричинити підвищення витрат електроенергії приблизно на 3–7 % залежно від дорожніх умов та стилю керування.

Суттєвий вплив на енергоспоживання має аеродинамічний опір кузова. При русі на швидкостях понад 70–80 км/год саме аеродинамічний фактор стає одним із головних джерел втрат енергії. Зі збільшенням швидкості сила опору повітря зростає непропорційно, що призводить до різкого підвищення навантаження на електропривід. У процесі аналізу було встановлено, що при русі зі швидкістю 90 км/год середнє споживання електроенергії може становити 15–17 кВт·год на 100 км, тоді як при швидкості 130 км/год цей показник часто перевищує 24–27 кВт·год на 100 км.

В процесі експлуатації електромобіля величина витрат електроенергії формується внаслідок одночасного впливу механічних, електричних, кліматичних та експлуатаційних факторів. Практичний аналіз показує, що навіть незначна зміна умов руху може призводити до суттєвого коливання показників енергоспоживання. Тому, для забезпечення ефективної роботи системи моніторингу необхідно враховувати максимально можливу кількість параметрів, які прямо чи опосередковано впливають на навантаження електроприводу.

Одним із головних факторів виступає швидкість руху транспортного засобу. В процесі збільшення швидкості суттєво підвищується аеродинамічний

опір, який прямо впливає на потужність, необхідну для підтримання руху автомобіля. На практиці це проявляється у значному зростанні споживаного струму при русі на автомагістралях. Наприклад, під час руху зі швидкістю 50–60 км/год система електроприводу може працювати у режимі відносно низького навантаження, тоді як при швидкості понад 120 км/год навантаження на інвертор, електродвигун та акумулятор збільшується майже у два рази.

Важливим фактором є характер прискорення автомобіля. Електродвигуни мають високий крутний момент уже з початкових обертів, що забезпечує динамічний розгін транспортного засобу. Але, саме інтенсивні прискорення спричиняють найбільші пікові навантаження на батарею. В моменти різкого натискання педалі акселератора система керування може короткочасно споживати десятки або навіть сотні кіловат потужності. Це призводить до збільшення тепловиділення в акумуляторі та підвищення втрат енергії.

На енергоспоживання суттєво впливає і дорожній рельєф. Під час руху вгору збільшується механічне навантаження на електродвигун, оскільки частина енергії витрачається на подолання сили тяжіння. На крутих підйомах потужність споживання може збільшуватися у 1,5–2 рази порівняно з рухом по рівній дорозі. Водночас, при русі вниз активується система рекуперативного гальмування, яка дозволяє частково повертати енергію назад до акумуляторної батареї.

Практичні вимірювання, також, підтверджують суттєвий вплив стану дорожнього покриття. Рух по мокрій, засніженій чи нерівній дорозі призводить до підвищення опору коченню коліс, через що збільшується навантаження на силову установку. Додатковий вплив створює тиск у шинах. При недостатньому тиску збільшується площа контакту шини з дорогою, що спричиняє додаткові втрати енергії.

Під час аналізу енергоспоживання електромобілів було встановлено, що режими руху є одним із найбільш критичних факторів, які визначають ефективність використання акумуляторної батареї. На практиці експлуатація транспортного засобу в міському циклі, на замських дорогах та автомагістралях формує абсолютно різні профілі навантаження електроприводу.

У міському режимі руху характерними є часті зупинки, інтенсивні прискорення та гальмування. У такому режимі електродвигун працює у динамічному циклі постійної зміни потужності. З одного боку, це підвищує миттєві витрати енергії під час розгону, проте одночасно забезпечує ефективну роботу системи рекуперації. Під час гальмування електродвигун переходить у генераторний режим та повертає частину енергії назад до батареї. У результаті реальне середнє енергоспоживання у місті часто виявляється нижчим, ніж при русі трасою на високій швидкості.

На заміських дорогах режим руху є більш стабільним. Електродвигун працює у режимі рівномірного навантаження, що позитивно впливає на стабільність споживання енергії. В такому випадку середнє енергоспоживання залежить переважно від швидкості руху та рельєфу місцевості.

Найбільше навантаження на акумуляторну батарею виникає під час руху автомагістралями. На високих швидкостях суттєво зростає аеродинамічний опір, а ефективність рекуперації практично знижується через відсутність частих гальмувань. Тому, запас ходу електромобіля при русі зі швидкістю 130–140 км/год може зменшуватися майже на третину порівняно з міським режимом.

Суттєвий вплив на енергоспоживання мають погодні умови. Низька температура навколишнього середовища призводить до зниження ефективності хімічних процесів у літій-іонних елементах. Крім цього, частина енергії витрачається на підігрів батареї для підтримання її робочого температурного діапазону. Влітку значний вплив створює система кондиціонування, особливо під час руху у міських заторах.

Для ефективного аналізу енергоспоживання електромобіля необхідно здійснювати постійний моніторинг великої кількості експлуатаційних параметрів, які характеризують роботу силової установки, акумуляторної батареї та допоміжних систем транспортного засобу. Ці дані виступають основою для подальшої обробки інформації програмною системою та формування аналітичних висновків.

Одним із основних параметрів є миттєва потужність споживання електроенергії. Даний показник дозволяє оцінити поточне навантаження на систему електроприводу та визначити режими найбільш інтенсивного використання енергії. Паралельно аналізується значення напруги акумуляторної батареї та струму навантаження. На основі цих параметрів можна визначити ефективність роботи батареї та оцінити рівень її навантаження під час різних режимів руху.

Важливим параметром виступає рівень заряду акумуляторної батареї, який зазвичай позначається як SOC (State of Charge). Даний показник дозволяє оцінювати залишковий запас енергії та прогнозувати можливу дальність руху транспортного засобу. В процесі моніторингу також аналізується температура акумуляторної батареї, так як перевищення допустимих температурних меж негативно впливає на ресурс елементів живлення.

До важливих параметрів належать швидкість руху автомобіля, прискорення, режим роботи електродвигуна та частота обертання коліс. На основі цих даних програмна система може визначати залежність між динамікою руху та витратами електроенергії.

Додатково система аналізує роботу рекуперативного гальмування. Під час переходу електродвигуна в генераторний режим визначається кількість енергії, яка повертається назад до акумуляторної батареї. Це дозволяє оцінити ефективність рекуперації в різних умовах експлуатації.

В межах програмної реалізації всі експлуатаційні дані можуть зчитуватися у вигляді часових рядів із подальшим записом в базу даних, або CSV-файли. Після цього програмна система виконує аналіз змін параметрів у часі, формує статистику середнього споживання, визначає пікові навантаження та будує графіки енергоспоживання.

1.2. Методи моніторингу та аналізу експлуатаційних даних

Практичний аналіз сучасних транспортних систем показує, що кількість параметрів, які можуть контролюватися під час роботи електромобіля, перевищує декілька сотень. До них належать показники швидкості руху, струму тягової батареї, напруги елементів живлення, температурних режимів, навантаження електродвигуна, режимів рекуперації, параметрів інвертора та характеристик допоміжних систем. Всі ці дані передаються через внутрішню цифрову мережу транспортного засобу та можуть використовуватися для побудови системи моніторингу енергоспоживання.

В процесі експлуатації електромобіля найбільш інтенсивний потік інформації формується саме системою керування батареєю BMS (Battery Management System). Дана система постійно виконує контроль стану акумуляторних елементів, аналізує рівень заряду, температуру, допустимі струми навантаження та рівень деградації батареї. На практиці BMS є одним із головних джерел телеметричних даних, оскільки саме вона дозволяє оцінити реальну ефективність використання електричної енергії.

В сучасних транспортних засобах для передачі телеметричної інформації найчастіше використовується цифрова шина CAN (Controller Area Network). Дана система забезпечує обмін інформацією між електронними блоками керування автомобіля з високою швидкістю та мінімальною затримкою. Через CAN-шину передаються дані про струм батареї, температуру силових модулів, швидкість обертання електродвигуна, положення педалі акселератора, параметри рекуперації та інші експлуатаційні характеристики. Тому, в більшості програмних систем моніторингу використовується підключення через OBD-II адаптери, або спеціалізовані CAN-інтерфейси.

В сучасних електромобілях система збору телеметричних даних фактично виступає окремим цифровим середовищем, у межах якого відбувається безперервний обмін інформацією між електронними модулями транспортного засобу. У процесі експлуатації автомобіля датчики та контролери постійно

генерують інформацію про роботу силових вузлів, параметри батареї, навантаження електроприводу та режими руху.

Основа системи збору даних складають електронні блоки керування ECU (Electronic Control Unit), які відповідають за обробку сигналів від датчиків та передачу інформації між окремими вузлами автомобіля. Кожен модуль виконує локальний контроль певної системи, після чого передає дані до центральної мережі транспортного засобу.

Одним із найбільш важливих компонентів є система BMS, яка забезпечує контроль акумуляторної батареї. В межах роботи даної системи виконується вимірювання напруги кожного елемента, температури батарейних модулів, струмів заряду та розряду, а також аналізується балансування елементів живлення. На основі отриманих даних система визначає рівень заряду батареї, прогнозує запас ходу та формує сигнали захисту у випадку перевищення допустимих параметрів.

Для передачі телеметричних даних між блоками керування найчастіше використовується CAN-шина. Її перевагою є висока стійкість до перешкод, надійність передачі даних та можливість роботи у режимі реального часу. Практичний аналіз показує, що сучасні електромобілі можуть передавати сотні параметрів із частотою оновлення від декількох разів на секунду до десятків повідомлень за секунду.

У процесі моніторингу експлуатаційних даних широкого поширення набули OBD-II інтерфейси, через які можна отримувати доступ до параметрів транспортного засобу. За допомогою Bluetooth або Wi-Fi адаптерів інформація передається на комп'ютер чи мобільний пристрій, де виконується її подальша обробка.

Практичне використання телеметричних систем дозволяє не лише аналізувати витрати електроенергії, а й виявляти аномальні режими роботи транспортного засобу. Для прикладу, різке збільшення температури батареї, або надмірний струм навантаження можуть свідчити про підвищене навантаження на силову установку чи несправність окремих компонентів.

Після отримання телеметричних даних важливим етапом є їхня обробка та подальший аналіз. В межах систем моніторингу енергоспоживання основна увага приділяється визначенню закономірностей між режимами руху транспортного засобу та рівнем витрат електричної енергії.

Одним із найбільш поширених методів виступає аналіз часових рядів. В цьому випадку кожен параметр фіксується у певний момент часу, після чого система визначає динаміку зміни характеристик під час руху автомобіля. Такий спосіб дозволяє оцінити вплив прискорень, швидкісних режимів та роботи допоміжних систем на загальне енергоспоживання.

Для аналізу ефективності руху широко використовується розрахунок середнього споживання електроенергії у кВт·год на 100 км. Даний показник дозволяє порівнювати режими експлуатації та визначати умови найбільш ефективного використання акумуляторної батареї.

У межах систем моніторингу, також, застосовується аналіз пікових навантажень. Під час різких прискорень, або руху вгору система фіксує короткочасне збільшення споживаної потужності. Подальший аналіз таких ділянок дозволяє оцінити навантаження на батарею та ефективність роботи електроприводу.

Важливе значення має фільтрація даних. У процесі зчитування телеметричної інформації можуть виникати короткочасні стрибки значень, похибки датчиків, або випадкові аномалії. Для усунення таких відхилень використовуються методи згладжування, ковзного середнього та цифрової фільтрації.

У процесі аналізу енергетичних показників важливим ще є визначення ефективності рекуперативного гальмування. Для цього система порівнює кількість енергії, яка витрачається під час прискорення, із кількістю енергії, що повертається до батареї під час гальмування.

В теперішніх системах моніторингу електромобілів програмне забезпечення виступає основним інструментом збору, обробки та візуалізації експлуатаційних даних. Програмна частина забезпечує автоматичний аналіз

великих масивів інформації та дозволяє оперативно визначати зміни у роботі транспортного засобу.

Таблиця 1.1 – Методи аналізу експлуатаційних даних електромобіля

Метод аналізу	Призначення
Аналіз часових рядів	Визначення змін параметрів у часі
Розрахунок середніх значень	Оцінка ефективності руху
Аналіз пікових навантажень	Виявлення максимального споживання
Фільтрація даних	Усунення шумів та похибок
Порівняльний аналіз режимів	Визначення оптимальних умов експлуатації
Побудова графіків	Візуальний аналіз параметрів

Для реалізації систем аналізу енергоспоживання широко використовується мова програмування Python, оскільки вона має велику кількість бібліотек для роботи з телеметричними даними, математичного аналізу та побудови графіків. В межах програмної реалізації можуть використовуватися бібліотеки Pandas для обробки таблиць даних, NumPy для числових розрахунків та Matplotlib для візуалізації параметрів енергоспоживання.

У процесі роботи система отримує дані від OBD-II адаптера, або збережених CSV-файлів, після чого виконується автоматичний аналіз параметрів. Програмне забезпечення обчислює середні значення споживання, визначає пікові навантаження, формує статистику витрат електроенергії та будує графіки зміни параметрів у часі.

Практичний аналіз показує, що візуалізація даних суттєво спрощує оцінку ефективності роботи електромобіля. Графіки зміни струму батареї, потужності електродвигуна або температури акумулятора дозволяють швидко виявляти аномальні режими експлуатації.

Окрему увагу приділено автоматизації формування звітів. Після завершення аналізу система може автоматично формувати таблиці середнього енергоспоживання, статистику рекуперації, оцінку запасу ходу та порівняння різних режимів руху. Це дозволяє значно спростити процес оцінки ефективності використання електричної енергії та підвищити інформативність результатів моніторингу.

1.3. Обґрунтування структури системи моніторингу

В рамках цієї роботи система розглядається як прикладне програмне рішення, призначене для приймання експлуатаційних даних, їх попередньої обробки, розрахунку енергетичних показників, виявлення змін у споживанні електроенергії та формування зрозумілих результатів у вигляді таблиць, графіків і підсумкових висновків. Основна увага приділяється не складній апаратній реалізації, а саме програмній логіці аналізу даних, оскільки головною метою є створення зручного інструмента, який дозволяє оцінювати енергоспоживання електромобіля на основі вже отриманих, або змодельованих експлуатаційних параметрів.

У практичному вигляді така система може працювати з файлами формату CSV чи Excel, у яких зберігаються дані про рух електромобіля, стан акумуляторної батареї, швидкість, пробіг, температуру, рівень заряду, спожиту енергію та рекуперацію. Такий підхід є зручним для дипломної роботи, оскільки не потребує обов'язкового підключення до реального автомобіля, але водночас дозволяє відтворити логіку реальної системи моніторингу. Програма може приймати набір експлуатаційних даних, виконувати їх перевірку, очищення, розрахунок ключових показників і будувати графіки, за якими можна оцінити, у яких режимах автомобіль споживає найбільше енергії.

Структура системи моніторингу повинна бути побудована таким чином, щоб кожен етап роботи програми був логічно відокремлений. Спочатку здійснюється завантаження даних, далі виконується їх перевірка та підготовка,

після цього розраховуються показники енергоспоживання, а на завершальному етапі результати подаються у вигляді таблиць, графіків і короткого аналітичного звіту. Така структура робить програму зрозумілою, зручною для тестування та придатною для подальшого розширення.

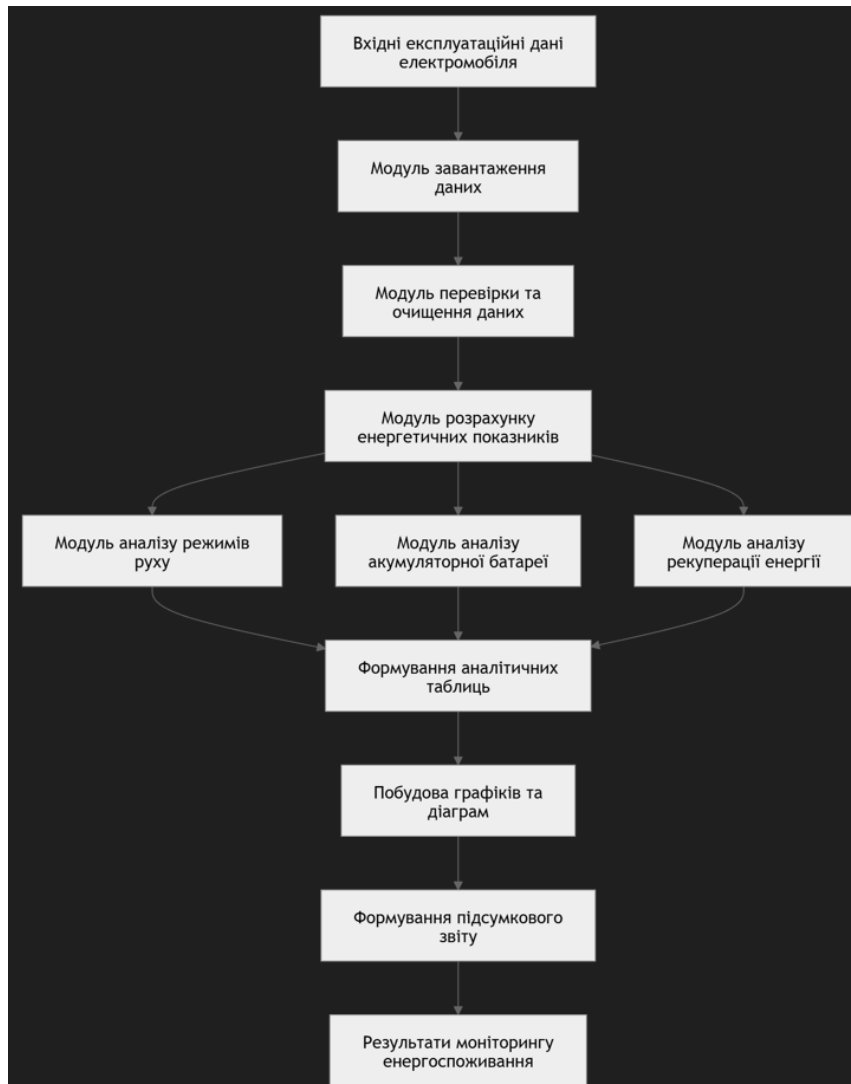


Рисунок 1.1 – Архітектура програмної системи моніторингу та аналізу енергоспоживання електромобіля

Запропонована структура дозволяє розділити систему на кілька функціональних модулів. Модуль завантаження даних відповідає за отримання вхідної інформації з файлу або іншого джерела. Модуль перевірки та очищення даних потрібен для виявлення пропущених значень, помилкових записів, різких

аномальних стрибків, або некоректних одиниць вимірювання. Модуль розрахунку енергетичних показників визначає середнє споживання електроенергії, витрати на 100 км, зміну рівня заряду батареї, пікові навантаження та частку рекуперованої енергії. Окремі аналітичні модулі дають змогу деталізувати оцінку за режимами руху, станом акумуляторної батареї та ефективністю повернення енергії під час гальмування.

Основною вимогою до системи є можливість роботи з експлуатаційними даними транспортного засобу. До таких даних належать час вимірювання, швидкість руху, пробіг, рівень заряду акумуляторної батареї, напруга, струм, температура батареї, миттєва потужність, енергія, витрачена на рух, та енергія, повернута під час рекуперації. Навіть якщо частина параметрів не зчитується з реального автомобіля, програма повинна мати структуру, яка дозволяє працювати з такими полями у вигляді таблиці.

Програма повинна автоматично перевіряти вхідні дані перед виконанням розрахунків. Це необхідно для того, щоб уникнути помилок в результатах аналізу. Наприклад, якщо у файлі відсутні значення швидкості, пробігу чи рівня заряду батареї, система має або повідомити про це користувача, або виконати коректну обробку пропусків. Якщо значення швидкості має нереалістичний характер, наприклад 400 км/год, програма повинна визначити такий запис як аномальний. Якщо рівень заряду батареї виходить за межі 0–100 %, такі дані також мають перевірятися.

Важливою вимогою є розрахунок ключових енергетичних показників. Система повинна визначати загальний пробіг за період аналізу, сумарне споживання електроенергії, середнє споживання у кВт·год/100 км, середню та максимальну швидкість, зміну рівня заряду акумулятора, кількість рекуперованої енергії, частку рекуперації та орієнтовний залишковий запас ходу. Ці показники дають змогу оцінити ефективність експлуатації електромобіля та встановити, за яких умов споживання енергії є найбільшим.

Окрему увагу потрібно приділити зручності використання програми. Система не повинна вимагати складного налаштування. Користувач має мати

можливість завантажити файл із даними, запустити програму та отримати результат у зрозумілому вигляді. Для дипломної роботи це особливо важливо, оскільки програма має демонструвати не лише технічну правильність розрахунків, а й практичну придатність для аналізу.

Ще однією важливою вимогою є наочне представлення результатів. Табличні дані дають змогу оцінити числові показники, але для швидкого розуміння динаміки енергоспоживання більш зручними є графіки. Тому система повинна будувати графік зміни швидкості, графік споживання електроенергії, графік зміни рівня заряду батареї та графік залежності витрат енергії від швидкості, або режиму руху. Це дозволить виявити пікові навантаження, періоди підвищеного споживання та ділянки, де рекуперація працювала найбільш ефективно.

З технічної точки зору система повинна бути побудована модульно. Це дозволяє окремо змінювати блок завантаження даних, блок розрахунків чи блок візуалізації без повного переписування програми. Наприклад, на початковому етапі система може працювати з CSV-файлами, а в подальшому її можна доповнити підключенням до OBD-II адаптера, бази даних чи хмарного сервісу.

Для реалізації системи моніторингу та аналізу енергоспоживання електромобіля доцільно використати мову програмування Python. Такий вибір є практично виправданим, так як Python добре підходить для обробки табличних даних, математичних розрахунків, побудови графіків та створення прикладних аналітичних програм. Python дозволяє швидко реалізувати програму, яка зчитує експлуатаційні дані, виконує їх обробку та формує результати аналізу у зручному вигляді.

Основним інструментом для роботи з табличними даними доцільно обрати бібліотеку Pandas. Вона дозволяє завантажувати CSV- або Excel-файли, працювати з рядками та стовпцями, виконувати фільтрацію, групування, сортування й обчислення середніх, мінімальних та максимальних значень. Для задачі аналізу енергоспоживання це особливо важливо, оскільки експлуатаційні

параметри електромобіля зазвичай зберігаються у вигляді таблиці, де кожен рядок відповідає конкретному моменту часу або окремому вимірюванню.

Для числових розрахунків може використовуватися бібліотека NumPy. Вона зручна для роботи з масивами числових значень, розрахунку середніх показників, обробки великих наборів даних та виконання базових математичних операцій. У практичній частині роботи NumPy може застосовуватися для розрахунку споживаної потужності, середнього енергоспоживання, відхилень від нормальних значень та інших числових характеристик.

Для побудови графіків доцільно використати бібліотеку Matplotlib. Вона дозволяє створювати лінійні графіки, діаграми, гістограми та комбіновані візуалізації. У контексті даної роботи це може бути графік зміни рівня заряду батареї у часі, графік залежності енергоспоживання від швидкості, графік рекуперації та графік зміни температури акумулятора. За потреби програма може бути доповнена бібліотекою Seaborn або Plotly, однак для базової реалізації достатньо Matplotlib, оскільки вона забезпечує просту та зрозумілу побудову основних графіків.

Вхідні дані доцільно зберігати у форматі CSV. Такий формат є простим, універсальним і зручним для обробки засобами Python. CSV-файл можна легко створити вручну, експортувати з електронної таблиці або отримати з іншої програми. Крім цього, використання CSV спрощує перевірку програмної системи, оскільки тестові експлуатаційні дані можна швидко змінювати та повторно запускати аналіз.

У разі розширення системи можна передбачити використання бази даних SQLite. Вона не потребує окремого серверного налаштування та добре підходить для локального зберігання результатів вимірювань. В такому випадку програма зможе не лише аналізувати один файл, а й накопичувати історію експлуатаційних даних за різні поїздки. Це дозволить порівнювати споживання електроенергії за днями, маршрутами, сезонами або різними режимами руху.

Загальна логіка вибору технологій полягає в тому, щоб система була простою, зрозумілою, доступною для запуску та придатною для подальшого

розвитку. Для дипломної роботи важливо, щоб програму можна було легко продемонструвати, пояснити її структуру та показати результати роботи на тестових, або реальних експлуатаційних даних.

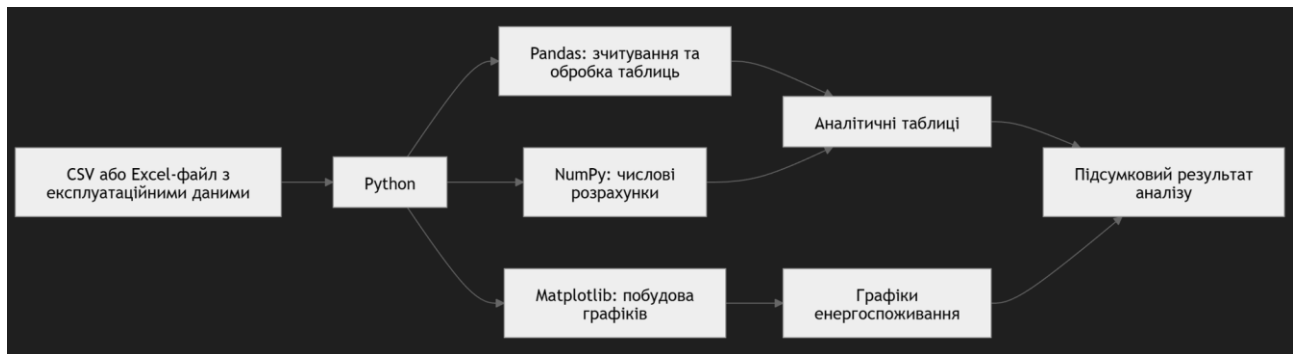


Рисунок 1.2 – Технологічна схема реалізації програмної системи аналізу енергоспоживання

Запропонована технологічна схема є достатньо простою для реалізації, але водночас охоплює всі основні етапи роботи з експлуатаційними даними. Python виступає центральним середовищем обробки, Pandas відповідає за роботу з таблицями, NumPy забезпечує розрахунки, а Matplotlib використовується для візуалізації. Такий набір інструментів дозволяє створити повноцінну систему моніторингу без надмірного ускладнення програмної частини.

Висновки до першого розділу

У першому розділі проведено аналіз особливостей енергоспоживання електромобілів та факторів, що впливають на витрати електричної енергії під час експлуатації транспортного засобу. Встановлено, що рівень енергоспоживання залежить від швидкості руху, режимів прискорення та гальмування, навантаження на електропривод, температурних умов та ефективності роботи системи рекуперації.

Досліджено сучасні методи збору телеметричних даних електромобілів, проаналізовано можливості використання OBD-інтерфейсів, CAN-шини та

систем керування батареєю для отримання експлуатаційної інформації. Визначено основні параметри, необхідні для моніторингу енергоспоживання та оцінювання технічного стану акумуляторної батареї.

На основі проведеного аналізу обґрунтовано структуру програмної системи моніторингу та аналізу енергоспоживання електромобіля, визначено вимоги до її функціональних можливостей та сформовано постановку задачі розробки програмного забезпечення.

РОЗДІЛ 2. РОЗРОБКА СИСТЕМИ МОНІТОРИНГУ ТА АНАЛІЗУ ЕНЕРГОСПОЖИВАННЯ ЕЛЕКТРОМОБІЛЯ

2.1. Проектування структури програмної системи

Проектування структури програмної системи моніторингу та аналізу енергоспоживання електромобіля виконувалося з урахуванням того, що вхідні експлуатаційні дані надходять не з одного універсального файлу, а з декількох окремих текстових файлів, які мають різну структуру та різне призначення. У розробленій програмі передбачено роботу з файлами типу pidint, pid, obd, calc, cells, celltemperatures та info. Кожен із цих файлів містить окрему частину експлуатаційної інформації: сирі PID/CAN-повідомлення, OBD-параметри автомобіля, розрахункові енергетичні показники, напруги комірок тягової батареї, температури батарейних модулів та службові повідомлення про процес підключення до OBD-адаптера. Тому структура програмної системи була спроектована за модульним принципом, де кожен тип даних обробляється окремим функціональним блоком, а після цього результати об'єднуються у загальну таблицю аналізу.

Основна логіка роботи системи полягає у послідовному виконанні кількох технічних етапів. Спочатку користувач завантажує набір TXT-файлів. Програма автоматично зчитує кожен файл, визначає його тип за назвою та структурою першого рядка, після чого передає файл у відповідний модуль обробки. Такий підхід дозволяє не задавати тип файлу вручну, а виконувати розпізнавання автоматично. Наприклад, якщо файл має заголовок Time;PID;hex0;hex1;..., система визначає його як PID-файл із байтами у шістнадцятковому форматі. Якщо файл має структуру Time;Parameter;Value, програма додатково аналізує назву файлу та визначає, чи це OBD-дані, чи розрахункові дані calc. Якщо файл містить поля Module, Cell, Volts, він обробляється як таблиця напруг комірок батареї.

Розроблена структура орієнтована на те, щоб програма могла працювати як із повним набором експлуатаційних файлів, так і з окремими файлами. Якщо користувач завантажує тільки PID-файл, система виконує технічне розшифрування CAN/PID-повідомлень і формує таблицю сирих байтів, числових значень та ASCII-представлення. Якщо разом із PID-файлами завантажено OBD- або CALC-файли, програма формує повну аналітичну таблицю енергоспоживання, у якій можна оцінити напругу батареї, струм, потужність, SOC, температуру, швидкість, дистанцію та питоми споживання електроенергії.

Загальна структура системи передбачає не просто читання файлів, а повний цикл обробки експлуатаційної інформації. На першому етапі виконується завантаження даних. На другому етапі система визначає тип кожного файлу. На третьому етапі відбувається розшифрування та нормалізація даних. На четвертому етапі окремі таблиці об'єднуються за часовою міткою. На п'ятому етапі виконується розрахунок енергетичних показників. На шостому етапі проводиться пошук аномальних режимів. Після цього система формує підсумкові таблиці, будує графіки та зберігає результати у форматах Excel і ZIP. Завдяки такій структурі програму можна використовувати не лише як демонстраційний код для дипломної роботи, а і як повноцінний інструмент первинного аналізу експлуатаційних даних електромобіля.

Загальна архітектура програмної системи побудована за принципом послідовної конвеєрної обробки даних. Це означає, що кожен файл проходить через визначений набір етапів: завантаження, ідентифікація, розшифрування, нормалізація, об'єднання, аналітична обробка та збереження результатів. Така архітектура є зручною для даної задачі, оскільки експлуатаційні дані електромобіля мають різну структуру, але всі вони прив'язані до єдиного ключового параметра – часу вимірювання. Якраз часова мітка використовується як основа для об'єднання даних з різних джерел у єдиний масив.

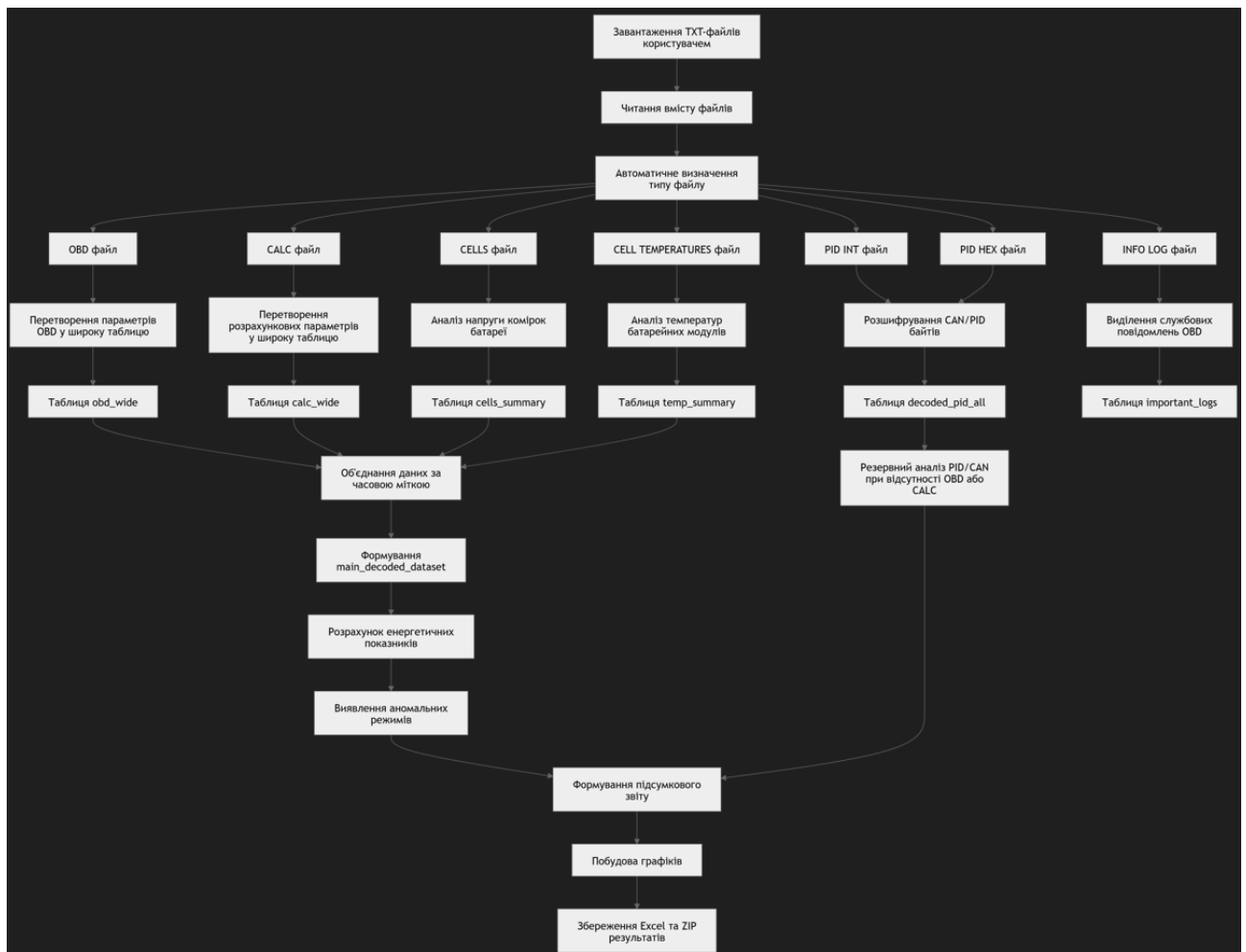


Рисунок 2.1 – Блок-схема архітектури програмної системи розшифрування та аналізу експлуатаційних даних електромобіля

На верхньому рівні система складається з модуля завантаження файлів, модуля визначення типу файлу, блоку розшифрування PID/CAN-повідомлень, блоку обробки параметричних OBD-даних, блоку обробки розрахункових енергетичних параметрів, блоку аналізу батарейних комірок, блоку аналізу температурних датчиків, блоку формування загального набору даних, блоку розрахунку енергетичних показників, блоку виявлення аномалій, блоку візуалізації та блоку експорту результатів.

Модуль завантаження файлів реалізований таким чином, щоб програму можна було запускати у середовищі Google Colab. Для цього використовується механізм завантаження файлів через стандартний інтерфейс Colab. Користувач може обрати один або декілька TXT-файлів, після чого програма зберігає їх у

робочому середовищі та формує список шляхів для подальшої обробки. У випадку запуску програми поза межами Colab передбачено резервний режим, у якому програма шукає TXT-файли у поточній папці.

Після завантаження даних працює модуль автоматичного визначення типу файлу. Його призначення полягає в тому, щоб зменшити участь користувача у налаштуванні програми. Програма самостійно аналізує перший рядок файлу та його назву. Якщо файл містить заголовок із байтами hex0–hex7, він передається у модуль розшифрування шістнадцяткових PID-даних. Якщо файл містить поля int0–int7, він обробляється як PID INT. Якщо файл має структуру Time;Parameter;Value, програма розпізнає його як OBD або CALC залежно від назви. Якщо у файлі присутні параметри модулів, комірок і напруги, він визначається як файл комірок батареї. Якщо у файлі вказані модуль, датчик і температура, він передається у температурний блок.

Центральним елементом архітектури є загальна таблиця main_decoded_dataset. Вона формується шляхом об'єднання OBD-даних, CALC-даних, зведеної таблиці напруг комірок і зведеної таблиці температур. Об'єднання виконується за найближчою часовою міткою з допустимим відхиленням у кілька секунд. Це важливо, оскільки різні файли можуть мати не повністю однакові часові позначки, але фактично належати до одного й того самого моменту роботи електромобіля. Такий підхід дозволяє не втрачати дані та формувати повніший набір параметрів для аналізу.

Після формування загальної таблиці система переходить до енергетичних розрахунків. Якщо у таблиці наявні напруга батареї BatteryV і струм батареї BatteryA, програма розраховує миттєву потужність батареї. Далі окремо визначається потужність розряду та потужність рекуперації. На основі різниці часу між сусідніми вимірюваннями програма розраховує крок енергії, сумарну витрачену енергію, рекуперовану енергію, пройдену дистанцію та питоме енергоспоживання у кВт·год/100 км. Завдяки цьому архітектура системи забезпечує не тільки розшифрування даних, а й реальний прикладний аналіз енергетичної ефективності електромобіля.

Окремою перевагою архітектури є наявність резервного режиму аналізу. Якщо користувач завантажив лише PID INT або PID HEX файл, але не завантажив OBD чи CALC, програма не завершує роботу з помилкою, а формує технічний звіт по CAN/PID-даних. У такому режимі виводиться кількість повідомлень, кількість унікальних PID, часовий діапазон запису та найчастіші PID-повідомлення. Це робить систему більш стійкою до неповного набору вхідних файлів і дозволяє демонструвати роботу програми навіть за мінімальної кількості даних.

Організація модулів збору та обробки даних у програмній системі побудована так, щоб кожен тип вхідної інформації мав власний алгоритм перетворення. Це дозволяє уникнути змішування різних форматів даних і забезпечує більш точну обробку кожного файлу. У програмі реалізовано окремі функціональні блоки, які відповідають за завантаження, читання, визначення типу, розшифрування, агрегування, об'єднання та аналітичну обробку даних.

Модуль завантаження даних відповідає за первинне отримання TXT-файлів. У середовищі Colab він використовує функцію завантаження файлів, після чого кожен отриманий файл зберігається у робочій директорії. Такий підхід дає змогу користувачу завантажити одразу всі файли, отримані з OBDZero або іншого інструмента збору даних. Програма не потребує ручного введення шляхів до файлів, що спрощує її використання під час демонстрації.

Модуль читання файлів виконує безпечно відкриття текстових даних із різними кодуваннями. Це важливо, оскільки файли можуть бути сформовані різними пристроями або програмами, а тому іноді відрізняються способом кодування тексту. Для цього у програмі передбачено послідовну спробу читання у форматах utf-8, cp1251 та latin1. Якщо стандартні варіанти не спрацьовують, застосовується режим читання з ігноруванням помилок. Завдяки цьому програма може обробляти навіть такі TXT-файли, які не відкриваються коректно з першої спроби.

Модуль визначення типу файлу аналізує структуру першого рядка та назву файлу. Саме він вирішує, у який обробник буде передано файл. Якщо це сирі

PID-дані у форматі INT або HEX, вони передаються у блок `decode_pid_table`. Якщо це OBD або CALC, вони передаються у блок `pivot_parameter_file`, який перетворює вертикальну структуру `Time;Parameter;Value` у широку таблицю, де кожен параметр стає окремим стовпцем. Це перетворення є критично важливим, тому що для подальших розрахунків набагато зручніше працювати з рядком, у якому в одному часовому зрізі присутні всі параметри автомобіля.

Модуль розшифрування PID/CAN-повідомлень виконує технічне перетворення байтів у зрозумілі числові поля. Для HEX-файлів кожне значення `hex0–hex7` переводиться із шістнадцяткового формату у десятковий байт. Для INT-файлів байти вже подані у десятковому форматі, тому програма лише приводить їх до числового типу. Далі з набору байтів формуються 16-бітні беззнакові та знакові значення, ASCII-представлення, сума байтів, середнє значення байтів і сирий рядок байтів у форматі HEX. Така обробка дозволяє не лише зберегти сирі CAN-дані, а й отримати технічні поля, які можна використати для подальшого аналізу або пошуку закономірностей між PID і реальними параметрами автомобіля.

Модуль обробки OBD-даних працює з параметрами, які вже мають зрозумілі назви, наприклад `Speed`, `BatteryV`, `BatteryA`, `BatteryT`, `SoC1`, `SoC2`, `MotorRPM`, `RestRange`, `MotorA`, `RegenA`, `Loadkg` та інші. Після перетворення у широку таблицю ці дані стають основою для формування загального датасету. Саме OBD-файл є головним джерелом параметрів для розрахунку енергоспоживання, тому що в ньому містяться напруга батареї, струм, швидкість, SOC і температура.

Модуль обробки CALC-файлу працює з розрахунковими параметрами, які вже були попередньо сформовані додатком або зовнішньою системою. До таких параметрів належать потужність, енергія, витрати у `Wh/km`, залишковий запас ходу, середня швидкість, допоміжне навантаження та інші показники. У програмі ці параметри отримують префікс `calc_`, щоб не змішуватися з первинними OBD-даними. Це дозволяє одночасно мати у загальній таблиці як первинні виміряні значення, так і розрахункові показники.

Окремий модуль призначений для обробки файлу комірок батареї. У цьому файлі кожен рядок містить інформацію про конкретний модуль, комірку, напругу, температуру та SOC. Оскільки таких записів для одного часу може бути багато, програма агрегує їх у зведену таблицю за часовою міткою. Для кожного моменту часу визначається мінімальна, максимальна та середня напруга комірок, а також різниця між максимальною і мінімальною напругою. Саме цей показник дозволяє оцінити розбаланс батареї. Також розраховується середній SOC комірок і температурні характеристики.

Модуль обробки температур батарейних модулів виконує подібне агрегування. Для кожного моменту часу програма визначає мінімальну, максимальну, середню температуру та різницю температур між датчиками. Це дає можливість оцінити рівномірність нагрівання батарейного блоку. Якщо різниця температур між модулями стає значною, це може свідчити про нерівномірне теплове навантаження або проблеми з охолодженням.

Після обробки всіх окремих файлів працює модуль об'єднання даних. Він використовує метод пошуку найближчої часової мітки, оскільки записи в різних файлах можуть бути сформовані з різницею в одну або декілька секунд. Для цього використовується об'єднання з часовим допуском. В результаті формується єдина таблиця, в якій для одного моменту часу доступні параметри OBD, CALC, комірок батареї та температурних датчиків.

Завершальним етапом є модуль аналітичної обробки. Він розраховує потужність батареї, енергію за кожен часовий крок, сумарну витрачену енергію, рекуперовану енергію, пройдену дистанцію та середнє споживання. Додатково програма класифікує режим руху за швидкістю: стоянка, повільний міський рух, міський або приміський режим, трасовий режим і високошвидкісний режим. Це дозволяє оцінити, у якому режимі автомобіль споживає найбільше електроенергії.

В межах розробленої програмної системи структура бази даних реалізована у вигляді набору взаємопов'язаних таблиць, які створюються під час виконання програми та зберігаються у підсумковому Excel-файлі. Такий підхід є

доцільним для даної дипломної роботи, оскільки дозволяє не розгортати окремий сервер бази даних, але водночас зберігати всі результати у впорядкованому табличному вигляді. Фактично Excel-файл виконує роль локальної бази експлуатаційних параметрів, де кожен аркуш відповідає окремому типу даних або окремому етапу обробки.

Основною таблицею системи є `main_decoded_dataset`. Вона формується після об'єднання всіх доступних джерел даних за часовою міткою. У цій таблиці зберігаються підготовлені до аналізу параметри електромобіля: швидкість, напруга батареї, струм батареї, температура батареї, рівень заряду SOC, залишковий запас ходу, параметри допоміжних систем, показники комірок, температурні характеристики модулів і розраховані енергетичні значення. Саме ця таблиця є центральною, оскільки на її основі виконуються всі подальші розрахунки та будується більшість графіків.

Для зберігання сирих PID/CAN-повідомлень використовується таблиця `pid_all_decoded`. Вона містить час запису, ідентифікатор PID, набір сирих байтів, ASCII-представлення, 16-бітні значення, суму байтів, середнє значення байтів і назву вихідного файлу. Така таблиця потрібна для технічної перевірки даних і може використовуватися для подальшого поглибленого аналізу CAN-повідомлень. Вона також корисна у випадку, коли OBD або CALC-файли відсутні, оскільки дозволяє сформувати хоча б базовий звіт по структурі PID-даних.

Таблиці `obd_decoded` і `calc_decoded` зберігають параметри, перетворені з вертикального формату `Time;Parameter;Value` у широкий формат. У таблиці `obd_decoded` містяться первинні експлуатаційні параметри автомобіля, які були зчитані з OBD-системи. У таблиці `calc_decoded` містяться розрахункові параметри, отримані з файлу CALC. Важливо, що розрахункові параметри отримують префікс `calc_`, тому їх можна відрізнити від первинних даних.

Таблиця `cells_raw` зберігає детальну інформацію про кожен комірку батареї. Вона містить час, номер модуля, літерне позначення комірки, напругу, інтерпольовану температуру, SOC і розрахункові ємнісні параметри. Оскільки

така таблиця може містити велику кількість рядків, для аналітичної обробки додатково формується таблиця `cells_summary`. У ній для кожного моменту часу зберігаються мінімальна, максимальна, середня напруга комірок, розбаланс напруги, середній SOC і температурні показники. Саме `cells_summary` підключається до основної таблиці `main_decoded_dataset`.

Таблиця `cell_temperatures_raw` містить первинні температурні значення з датчиків батарейних модулів. Для кожного запису вказується час, номер модуля, номер сенсора та температура. На її основі формується таблиця `cell_temperatures_summary`, де для кожного моменту часу зберігаються мінімальна, максимальна, середня температура та температурна різниця між модулями. Ці дані використовуються для оцінки теплової рівномірності батареї.

Окремо зберігається таблиця `summary`, яка виконує роль підсумкового аналітичного звіту. В ній містяться загальна кількість записів, початок і кінець запису, тривалість вимірювання, середня та максимальна швидкість, середня напруга батареї, максимальний струм розряду, максимальна потужність, витрачена енергія, рекуперована енергія, розрахована дистанція, середнє споживання, зміна SOC і температурні характеристики. Така таблиця є зручною для вставлення у дипломну роботу як результат роботи програмної системи.

Таблиця `anomalies` використовується для збереження виявлених відхилень. У ній фіксується час події, тип аномалії та значення параметра. Програма може виявляти аномальну напругу батареї, надмірний струм, небажану температуру батареї, підвищений розбаланс комірок або значну нерівномірність температури модулів. Наявність такої таблиці підвищує практичну цінність системи, оскільки вона не лише рахує середні показники, а й дозволяє знаходити потенційно проблемні режими роботи.

Структура бази даних побудована так, щоб кожна таблиця мала чітке призначення. Сирі дані зберігаються окремо від підготовлених даних, зведені таблиці окремо від детальних, а підсумкові результати окремо від технічних логів. Це спрощує перевірку роботи програми та дозволяє швидко знайти потрібний рівень деталізації. Якщо потрібно перевірити конкретний CAN-пакет,

використовується таблиця PID. Якщо потрібно оцінити енергоспоживання, використовується `main_decoded_dataset` чи `summary`. Якщо потрібно проаналізувати батарею, використовуються `cells_summary` і `cell_temperatures_summary`.

В результаті розроблена структура бази експлуатаційних параметрів забезпечує повний цикл збереження даних: від сирих TXT-файлів до підсумкового аналітичного звіту. Такий підхід є зручним для дипломної роботи, оскільки дозволяє продемонструвати не тільки саму програму, а й логіку організації даних, порядок їх обробки та технічну послідовність формування результатів аналізу.

2.2. Розробка програмного забезпечення системи

Розробка програмного забезпечення системи моніторингу та аналізу енергоспоживання електромобіля виконувалася мовою програмування Python із використанням бібліотек Pandas, NumPy, Matplotlib та OpenPyXL. Основною вимогою до програмного забезпечення була можливість автоматичного оброблення експлуатаційних даних, отриманих із декількох джерел, їх синхронізація за часовими мітками, виконання енергетичних розрахунків та формування підсумкових звітів без ручного втручання користувача.

Програмна система реалізована як набір взаємопов'язаних функціональних модулів, кожен із яких виконує окремий етап обробки інформації. Загальний обсяг програмного коду перевищує тисячу рядків та включає модулі завантаження файлів, визначення типу даних, розшифрування CAN/PID-повідомлень, обробки OBD-параметрів, аналізу батарейних комірок, розрахунку енергоспоживання, виявлення аномалій, побудови графіків та експорту результатів.

В основі роботи програмного забезпечення лежить автоматичний аналіз структури вхідних TXT-файлів. Користувач завантажує набір файлів, після чого система самостійно визначає їх тип та виконує відповідний алгоритм обробки.

Такий підхід дозволяє мінімізувати кількість ручних налаштувань та підвищити зручність використання програми.

Програмна система підтримує обробку сирих CAN/PID-даних, параметрів OBD, розрахункових характеристик, інформації про стан батарейних комірок та температурних сенсорів. Після завершення аналізу користувач отримує структурований набір результатів у форматі Excel, ZIP-архів із графіками та аналітичний звіт про стан електромобіля.

В процесі розробки особлива увага приділялася стійкості програмного забезпечення до неповних, або пошкоджених даних. Якщо частина файлів відсутня, система переходить у резервний режим роботи та виконує аналіз доступної інформації. Це дозволяє використовувати програму як для повного аналізу енергоспоживання, так і для технічного аналізу окремих типів даних.

Модуль зчитування експлуатаційних даних є початковою ланкою програмної системи та відповідає за завантаження, відкриття та підготовку вхідних файлів до подальшої обробки. Реалізація даного модуля виконана засобами Python із використанням стандартних бібліотек `os`, `io`, `re`, `zipfile`, а також бібліотеки `pandas`, яка забезпечує роботу з табличними структурами даних.

Для забезпечення сумісності із середовищем Google Colab реалізовано функцію `upload_files()`, яка використовує модуль `google.colab.files`. Після завантаження файлів їхній вміст записується у робочу директорію, а програма формує список шляхів до кожного файлу для подальшої обробки. У разі запуску програми поза середовищем Colab передбачено альтернативний механізм пошуку всіх TXT-файлів у поточній директорії.

Однією з важливих задач модуля є коректне читання текстових файлів із різними кодуваннями. Для цього реалізовано функцію `read_text_file()`, яка послідовно перевіряє кодування UTF-8, CP1251 та Latin-1. Якщо жодне кодування не підходить, використовується режим читання з ігноруванням помилок. Такий підхід забезпечує стабільну роботу програми з файлами, сформованими різними програмними комплексами та мобільними додатками.

Після зчитування виконується автоматичне визначення типу файлу функцією `detect_file_type()`. Визначення здійснюється на основі аналізу першого рядка файлу та його назви. Якщо виявлено заголовок `Time;PID;hex`, файл класифікується як `pid_hex`. Якщо знайдено структуру `Time;PID;int`, файл визначається як `pid_int`. Для параметричних файлів використовується аналіз поля `Parameter`, а для батарейних файлів – наявність колонок `Module`, `Cell`, `Volts` або `Temperature`.

Після класифікації дані зчитуються у структури `DataFrame`. Для більшості файлів використовується функція `read_semicolon_table()`, яка читає таблиці з роздільником `;`. Далі виконується автоматичне перетворення часових міток у формат `datetime` за допомогою функції `parse_time_column()`. Це забезпечує можливість подальшого сортування та синхронізації даних за часом.

Важливою особливістю модуля є автоматичне перетворення текстових значень у числові типи даних. Для цього використовується функція `safe_numeric()`, яка перетворює значення у формат `float` або `int`, а некоректні записи автоматично замінює на `NaN`. Це дозволяє уникнути аварійного завершення програми при наявності помилкових записів у файлах.

В результаті роботи модуля формується набір стандартизованих таблиць даних, придатних для подальшого аналізу. Завдяки цьому система може працювати з файлами різної структури та автоматично адаптуватися до їхнього формату без додаткового налаштування з боку користувача.

Головним елементом програмної системи є алгоритм аналізу витрат електроенергії, реалізований у функції `calculate_energy_metrics()`. Даний алгоритм використовує експлуатаційні параметри електромобіля для визначення потужності батареї, спожитої енергії, рекуперації та питомого енергоспоживання.

На першому етапі алгоритм виконує сортування записів за часовою міткою та перевірку типів даних. Це необхідно для забезпечення коректного порядку обчислень і усунення можливих помилок під час математичної обробки.

Розрахунок миттєвої потужності батареї виконується за формулою:

$$P = \frac{U \cdot I}{1000} \quad (2.1)$$

де P – потужність батареї, кВт; U – напруга батареї, В; I – струм батареї, А.

Якщо струм має від’ємне значення, система інтерпретує його як режим розряду батареї. Якщо струм додатний, він розглядається як режим рекуперації енергії. Це дозволяє окремо розраховувати витрачену та повернену енергію.

Для визначення енергії використовується різниця часу між сусідніми вимірюваннями. Часовий інтервал переводиться в години, після чого для кожного кроку визначається приріст енергії:

$$E = P \cdot \Delta t \quad (2.2)$$

де E – енергія, кВт·год; P – потужність, кВт; Δt – часовий інтервал, год.

Отримані значення накопичуються за допомогою кумулятивної суми, у результаті чого формується загальна кількість витраченої та рекуперованої енергії протягом поїздки.

Розрахунок пройденої відстані здійснюється на основі швидкості руху автомобіля:

$$S = v \cdot \Delta t \quad (2.3)$$

де S – відстань, км; v – швидкість, км/год.

На основі накопиченої енергії та пройденої дистанції визначається питоме енергоспоживання:

$$W = \frac{E}{S} \cdot 100 \quad (2.4)$$

де W – споживання електроенергії, кВт·год/100 км.

Алгоритм також виконує автоматичну класифікацію режимів руху за швидкістю автомобіля. При швидкості 0 км/год фіксується режим стоянки. Інтервал від 0 до 30 км/год відповідає повільному міському руху. Швидкість від 30 до 70 км/год класифікується як міський або приміський режим. Діапазон від 70 до 110 км/год визначає трасовий режим, а швидкість понад 110 км/год вважається високошвидкісним режимом.

Додатково алгоритм реалізує функцію автоматичного виявлення аномалій. Програма аналізує напругу батареї, струм, температуру та параметри батарейних комірок. Якщо напруга виходить за допустимі межі, температура перевищує встановлені значення, або спостерігається значний розбаланс комірок, формується запис у таблиці anomalies. Це дозволяє використовувати систему не лише для аналізу енергоспоживання, а й для діагностики технічного стану електромобіля.

Візуалізація результатів є важливим елементом програмної системи, так як дозволяє користувачу швидко оцінити характер зміни параметрів електромобіля та виявити аномальні режими роботи. Для побудови графіків використовується бібліотека Matplotlib, яка забезпечує створення високоякісних графічних матеріалів.

У програмі реалізовано окрему функцію plot_results(), яка автоматично перевіряє наявність необхідних параметрів та формує відповідні графіки. Якщо у наборі даних присутня швидкість руху, будується графік зміни швидкості електромобіля у часі. При наявності даних про потужність батареї формується графік миттєвої потужності. Для аналізу стану батареї створюються графіки зміни рівня заряду SOC та температури акумулятора.

Додатково система будує графік питомого енергоспоживання у функції пройдені дистанції. Це дозволяє оцінити, як змінюється ефективність використання електроенергії під час руху. Окремо реалізовано гістограму розподілу витраченої енергії за режимами руху. Такий підхід дає змогу

визначити, в яких умовах автомобіль споживає найбільшу кількість електроенергії.

Для аналізу батареї формується графік розбалансу напруг комірок, який дозволяє оцінити рівномірність роботи акумуляторного блоку. Збільшення різниці між максимальною та мінімальною напругою комірок може свідчити про початок деградації батареї або нерівномірне навантаження окремих модулів.

Формування статистики реалізовано функцією `build_summary()`. Вона автоматично обчислює кількість записів, тривалість вимірювання, середню та максимальну швидкість, середню напругу батареї, струм, потужність, витрачену та рекуперовану енергію, дистанцію, середнє споживання електроенергії та температурні характеристики батареї. Результати зберігаються в таблиці `summary`, яка використовується для формування підсумкового звіту.

Для збереження результатів використовується функція `save_outputs()`. Всі сформовані таблиці записуються в Excel-файл за допомогою бібліотеки `OpenPyXL`, а графіки автоматично зберігаються у форматі PNG. Після цього результати архівуються у ZIP-файл, який може бути завантажений користувачем або використаний для подальшого аналізу.

Отже, реалізований модуль візуалізації та статистичної обробки забезпечує не лише відображення результатів аналізу в графічному вигляді, але й автоматичне формування повного набору аналітичних матеріалів, необхідних для оцінювання енергоспоживання електромобіля та технічного стану його акумуляторної системи.

2.3. Реалізація функцій моніторингу енергоспоживання

Функції моніторингу енергоспоживання реалізовані в програмній системі на основі автоматичного аналізу експлуатаційних даних електромобіля, що надходять із OBD-файлів, розрахункових файлів CALC та інформації про стан тягової батареї. Програмна реалізація забезпечує безперервний аналіз параметрів руху, розрахунок енергетичних показників, визначення режимів експлуатації,

виявлення аномалій та формування підсумкових результатів у графічному й табличному вигляді.

Основою моніторингу є формування інтегрованого набору даних `main_decoded_dataset`, у якому кожен запис відповідає певному моменту часу та містить повний набір параметрів: швидкість руху, напругу батареї, струм, температуру, рівень заряду SOC, параметри комірок батареї та розрахункові характеристики. Використання єдиної часової шкали дозволяє аналізувати взаємозв'язок між режимами експлуатації автомобіля та його енергоспоживанням.

У програмі реалізовано автоматичне визначення режимів руху, обчислення витраченої та рекуперованої енергії, оцінку ефективності використання акумуляторної батареї та аналіз технічного стану силової системи. Результати розрахунків зберігаються у таблицях, що дозволяє проводити подальший аналіз та формувати звітність.

Однією з основних функцій програмної системи є визначення середнього енергоспоживання електромобіля в різних режимах експлуатації. Для цього в програмі реалізовано автоматичну класифікацію режимів руху на основі поточної швидкості автомобіля. Алгоритм розподіляє записи на режими стоянки, повільного міського руху, міського або приміського режиму, трасового руху та високошвидкісного режиму.

Для кожного режиму система накопичує значення витраченої енергії та визначає її середнє значення. Такий підхід дозволяє оцінити, у яких умовах експлуатації електромобіль споживає найбільшу кількість електроенергії. Наприклад, у міському режимі спостерігається часта зміна швидкості та активне використання рекуперації, тоді як у високошвидкісному режимі основний вплив на енергоспоживання створює аеродинамічний опір.

У програмній реалізації для кожного часового інтервалу обчислюється витрачена енергія та накопичується її сумарне значення. Далі визначається питоме енергоспоживання у кВт·год на 100 км, що дозволяє порівнювати ефективність руху за різних умов експлуатації. В результаті користувач отримує

кількісну оцінку енергетичної ефективності електромобіля залежно від режиму руху.

Для аналізу впливу швидкості на витрати електроенергії програмна система використовує дані про швидкість руху, струм та напругу батареї. На основі цих параметрів визначається миттєва потужність силової установки та її зміна у часі. Результати аналізу показують, що зі збільшенням швидкості зростає середнє навантаження на акумуляторну батарею та збільшується питоме енергоспоживання.

Програма автоматично розраховує пройдену відстань та накопичену енергію, що дозволяє визначити залежність між режимом руху та енергетичними витратами. Окремо враховується режим рекуперації, у якому частина енергії повертається до батареї під час гальмування.

Важливим елементом моніторингу є аналіз навантаження на батарею. Для цього використовується інформація про струм розряду та заряджання, температуру батареї та рівень заряду SOC. Значні значення струму або підвищення температури можуть свідчити про роботу електромобіля у режимі підвищеного навантаження. Програмна система автоматично фіксує такі ситуації та вносить їх до таблиці аномалій.

Крім цього, аналізується стан батарейних комірок. Визначається різниця між максимальною та мінімальною напругою комірок, що дозволяє оцінити ступінь їх балансування. Значний розбаланс може свідчити про нерівномірне старіння або деградацію окремих елементів акумуляторної батареї.

Після завершення обробки експлуатаційних даних програмна система автоматично формує набір звітів та графічних матеріалів. Основою звітності є таблиця *summary*, яка містить інтегральні показники роботи електромобіля за весь період спостереження.

У підсумковому звіті відображаються кількість записів, тривалість вимірювання, середня та максимальна швидкість руху, середня напруга батареї, максимальний струм розряду, витрачена та рекуперована енергія, пройдена

дистанція та середнє енергоспоживання. Додатково виводяться температурні характеристики батареї та зміна рівня заряду SOC.

Для візуального аналізу програма автоматично формує графіки швидкості руху, потужності батареї, температури акумулятора, зміни рівня заряду та питомого енергоспоживання. Окремо будується графік розбалансу комірок батареї, який дозволяє оцінити рівномірність роботи акумуляторного блока.

Програмна система також реалізує автоматичне формування таблиці anomalies, у якій реєструються перевищення допустимих меж напруги, струму, температури та параметрів батарейних модулів. Наявність такого механізму підвищує практичну цінність системи та дозволяє використовувати її не лише для аналізу енергоспоживання, а й для технічної діагностики електромобіля.

Результати роботи програми зберігаються у форматі Excel, де кожен аркуш відповідає окремому набору даних, або результату обробки. Графічні матеріали експортуються у формат PNG, а повний набір результатів автоматично архівується у ZIP-файл. Така організація вихідних даних забезпечує зручне використання результатів під час подальшого аналізу, документування та підготовки звітної документації.

Висновки до другого розділу

У другому розділі розроблено структуру програмної системи моніторингу та аналізу енергоспоживання електромобіля. Сформовано архітектуру програмного забезпечення, яка забезпечує автоматичне завантаження, розпізнавання, обробку та аналіз експлуатаційних даних, отриманих із різних джерел.

Реалізовано програмне забезпечення мовою Python із використанням бібліотек Pandas, NumPy, Matplotlib та OpenPyXL. Розроблено модулі зчитування TXT-файлів, розшифрування CAN/PID-повідомлень, обробки OBD-параметрів, аналізу стану батареї, розрахунку енергетичних показників та виявлення аномалій.

У програмі реалізовано функції визначення витраченої та рекуперованої енергії, розрахунку питомого енергоспоживання, аналізу режимів руху та формування підсумкових звітів і графічних матеріалів. Розроблена система забезпечує комплексний аналіз експлуатаційних даних електромобіля та формування результатів у зручному для користувача вигляді.

РОЗДІЛ 3. ДОСЛІДЖЕННЯ ЕФЕКТИВНОСТІ РОЗРОБЛЕНОЇ СИСТЕМИ

3.1. Тестування програмної системи

Тестування програмної системи моніторингу та аналізу енергоспоживання електромобіля проводилося з метою перевірки коректності роботи алгоритмів обробки експлуатаційних даних, розрахунку енергетичних показників, формування графічних матеріалів та виявлення аномальних режимів роботи тягової батареї. Розроблена система реалізована мовою Python, що дозволяє виконувати обробку реальних TXT-файлів, отриманих із системи діагностики електромобіля.

Під час тестування використовувався набір експлуатаційних даних, який містив інформацію про швидкість руху автомобіля, параметри батареї, рівень заряду, температуру, стан батарейних комірок та часові мітки вимірювань. Всі файли автоматично завантажувалися до програмної системи, після чого виконувалося їх розпізнавання, обробка та формування інтегрованого набору даних.

Результати тестування підтвердили працездатність розробленої програмної системи та можливість її використання для моніторингу енергоспоживання електромобілів у реальних умовах експлуатації.

Для перевірки працездатності системи використовувався набір реальних експлуатаційних даних електромобіля, отриманих у процесі руху транспортного засобу. Дані були збережені у текстових файлах формату TXT та містили параметри OBD, інформацію про батарею, батарейні комірки та результати розрахункових модулів.

Перед початком аналізу програмна система автоматично виконувала завантаження файлів, визначення їхнього типу та формування структур даних у форматі DataFrame. Вхідні параметри містили часові мітки, швидкість руху,

напругу батареї, струм, рівень заряду SOC, температуру батареї та характеристики окремих комірок акумуляторного блока.

За результатами тестового запуску було сформовано підсумковий аналітичний звіт, який містить інтегральні показники роботи електромобіля. Згідно з отриманими результатами, система обробила 538 записів протягом інтервалу 19,33 хвилини. Середня швидкість руху становила 34,89 км/год, максимальна швидкість досягала 86 км/год. Середня напруга тягової батареї становила 336,02 В, а діапазон зміни напруги знаходився у межах від 285,7 В до 357,6 В. Витрачена електроенергія за час тестування склала 1,3815 кВт·год, тоді як рекуперована енергія становила 0,2875 кВт·год. Середнє енергоспоживання системою було визначене на рівні 12,59 кВт·год/100 км, що відповідає типовим показникам сучасних електромобілів.

ПІДСУМКОВИЙ АНАЛІТИЧНИЙ ЗВІТ	
Показник	Значення
Кількість записів	538
Початок запису	2021-11-03 12:39:46.881000
Кінець запису	2021-11-03 12:59:06.683000
Тривалість запису, хв	19.33
Середня швидкість, км/год	34.89
Максимальна швидкість, км/год	86.0
Середня напруга батареї, В	336.02
Мінімальна напруга батареї, В	283.7
Максимальна напруга батареї, В	357.6
Середній струм батареї, А	-10.98
Максимальний струм розряду, А	131.81
Максимальний струм рекуперації, А	57.94
Середня потужність батареї, кВт	-3.477
Максимальна потужність розряду, кВт	37.394
Максимальна потужність рекуперації, кВт	20.719
Витрачена енергія, кВт·год	1.3815
Рекуперована енергія, кВт·год	0.2875
Розрахована дистанція, км	10.9722
Середнє споживання, кВт·год/100 км	12.591
SOC початковий, %	55.0
SOC кінцевий, %	45.0
Зміна SOC, %	10.0
Середня температура батареї, °C	7.78
Мінімальна температура батареї, °C	7.0
Максимальна температура батареї, °C	8.9
Максимальний розбаланс комірок, В	0.24

Рисунок 3.1 – Підсумковий аналітичний звіт програмної системи

Окремо було виконано перевірку алгоритму виявлення аномалій. Під час тестування система автоматично виявила випадки підвищеного розбалансу напруг батарейних комірок. Максимальне значення різниці напруг між комірками становило 0,24 В, що перевищує встановлений поріг контролю 0,08 В. Усі виявлені події були автоматично занесені до таблиці аномалій із фіксацією часу виникнення та значення параметра.

Виявлені потенційні аномалії:

Time	Anomaly	Value
2021-11-03 12:41:15.779	Підвищений розбаланс комірок	батареї 0.080
2021-11-03 12:42:28.387	Підвищений розбаланс комірок	батареї 0.190
2021-11-03 12:42:43.180	Підвищений розбаланс комірок	батареї 0.090
2021-11-03 12:44:43.984	Підвищений розбаланс комірок	батареї 0.120
2021-11-03 12:45:11.455	Підвищений розбаланс комірок	батареї 0.080
2021-11-03 12:46:35.880	Підвищений розбаланс комірок	батареї 0.090
2021-11-03 12:46:39.978	Підвищений розбаланс комірок	батареї 0.095
2021-11-03 12:54:33.483	Підвищений розбаланс комірок	батареї 0.080
2021-11-03 12:54:43.981	Підвищений розбаланс комірок	батареї 0.085
2021-11-03 12:54:47.978	Підвищений розбаланс комірок	батареї 0.085
2021-11-03 12:54:52.356	Підвищений розбаланс комірок	батареї 0.085
2021-11-03 12:54:54.386	Підвищений розбаланс комірок	батареї 0.090
2021-11-03 12:54:56.481	Підвищений розбаланс комірок	батареї 0.085
2021-11-03 12:55:06.981	Підвищений розбаланс комірок	батареї 0.100
2021-11-03 12:55:09.181	Підвищений розбаланс комірок	батареї 0.100
2021-11-03 12:55:13.280	Підвищений розбаланс комірок	батареї 0.085
2021-11-03 12:55:15.377	Підвищений розбаланс комірок	батареї 0.155
2021-11-03 12:55:17.576	Підвищений розбаланс комірок	батареї 0.085
2021-11-03 12:55:19.681	Підвищений розбаланс комірок	батареї 0.090
2021-11-03 12:55:21.782	Підвищений розбаланс комірок	батареї 0.125
2021-11-03 12:55:23.783	Підвищений розбаланс комірок	батареї 0.110
2021-11-03 12:55:26.085	Підвищений розбаланс комірок	батареї 0.105
2021-11-03 12:55:28.079	Підвищений розбаланс комірок	батареї 0.090
2021-11-03 12:55:30.184	Підвищений розбаланс комірок	батареї 0.115
2021-11-03 12:55:32.086	Підвищений розбаланс комірок	батареї 0.105
2021-11-03 12:55:34.231	Підвищений розбаланс комірок	батареї 0.105
2021-11-03 12:55:40.482	Підвищений розбаланс комірок	батареї 0.090
2021-11-03 12:56:14.189	Підвищений розбаланс комірок	батареї 0.130
2021-11-03 12:56:18.478	Підвищений розбаланс комірок	батареї 0.130
2021-11-03 12:56:20.591	Підвищений розбаланс комірок	батареї 0.240

Рисунок 3.2 – Результати автоматичного виявлення аномалій батарейного блока

Отже, підготовлені тестові дані дозволили перевірити роботу всіх функціональних модулів програмної системи та підтвердили її здатність працювати з реальними експлуатаційними файлами електромобіля.

Перевірка коректності роботи системи проводилася шляхом аналізу сформованих графіків та зіставлення отриманих результатів із фактичними параметрами руху електромобіля. В процесі тестування було підтверджено

правильність синхронізації часових міток та коректність об'єднання даних із різних джерел.

Аналіз графіка швидкості на Рис. 3.3 показав, що автомобіль рухався у змішаному режимі експлуатації. Спостерігалися ділянки прискорення, руху зі сталою швидкістю та повної зупинки транспортного засобу. Максимальна швидкість досягала близько 86 км/год, що відповідає значенню, отриманому в підсумковому звіті.

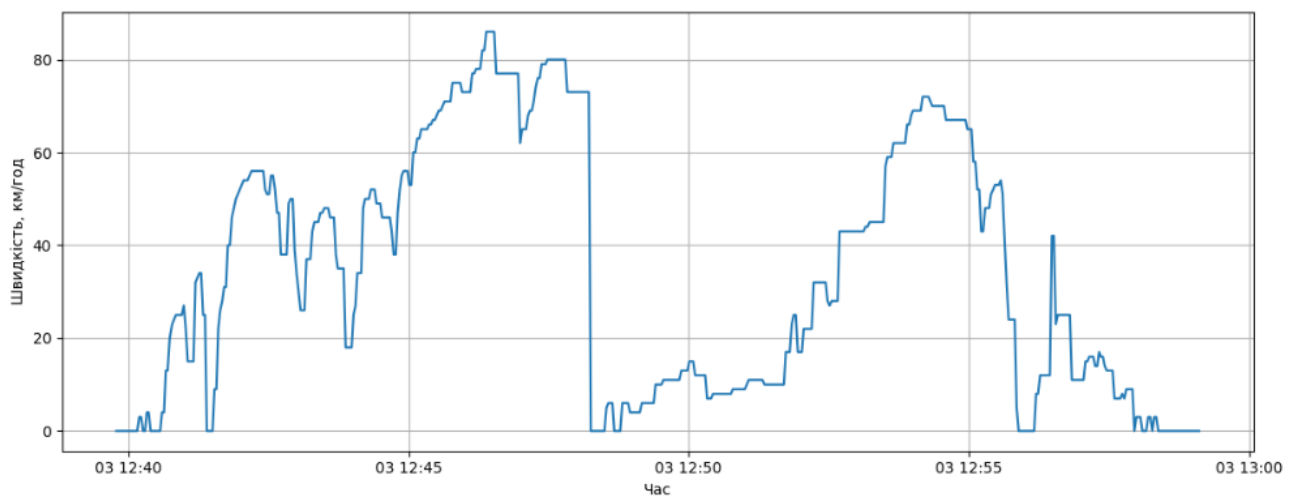


Рисунок 3.3 – Графік зміни швидкості електромобіля

Графік миттєвої потужності батареї на Рис. 3.4 підтвердив коректність роботи алгоритму розрахунку потужності. Від'ємні значення відповідають режиму розряду батареї, тоді як додатні значення свідчать про роботу системи рекуперації під час гальмування. Максимальна потужність розряду склала 37,39 кВт, а максимальна потужність рекуперації – 20,72 кВт.

Перевірка зміни рівня заряду на Рис. 3.5 показала поступове зменшення SOC від 55 % до 45 %, що відповідає витраті електроенергії під час руху. Характер зміни заряду має плавний вигляд без різких стрибків, що свідчить про коректність роботи алгоритму обробки батарейних параметрів.

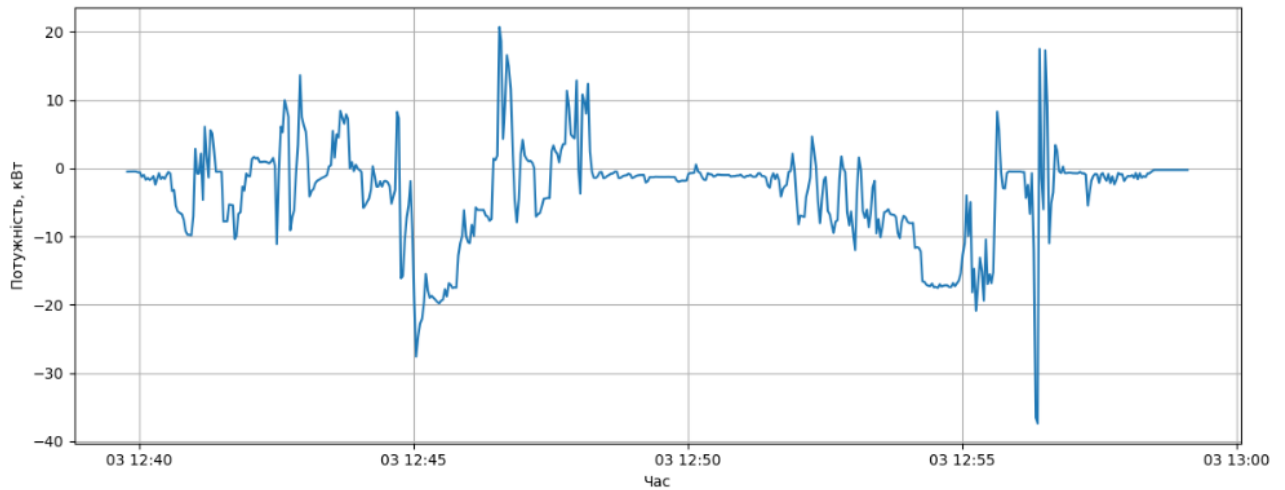


Рисунок 3.4 – Графік миттєвої потужності тягової батареї

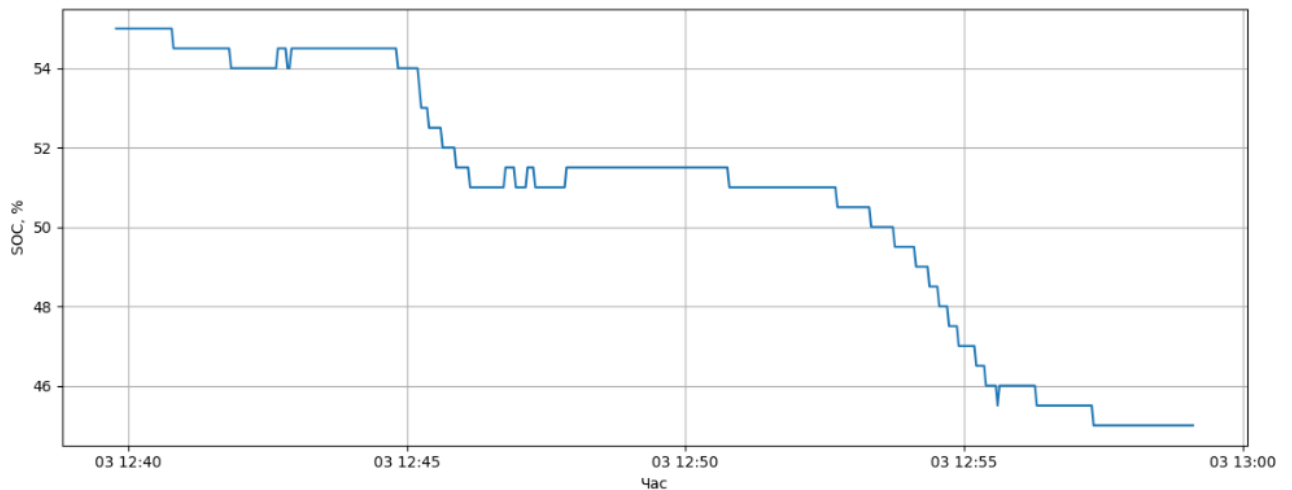


Рисунок 3.5 – Графік зміни рівня заряду тягової батареї

Аналіз температурного режиму на Рис. 3.6 показав поступове збільшення температури батареї від 7 °C до 8,9 °C у процесі експлуатації. Такі значення відповідають нормальному тепловому режиму роботи акумуляторної системи та не виходять за допустимі межі.

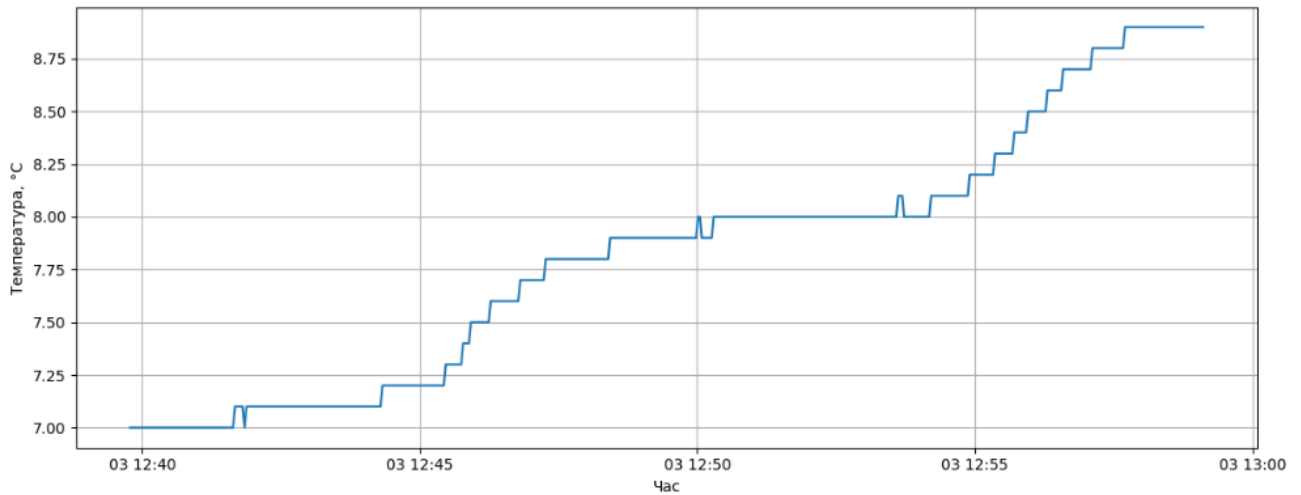


Рисунок 3.6 – Графік зміни температури батареї

Графік питомого енергоспоживання на Рис. 3.7 показав стабілізацію значення після початкового перехідного процесу та досягнення середнього рівня близько 12,6 кВт·год/100 км. Це підтверджує коректність реалізації алгоритму розрахунку енергетичної ефективності.

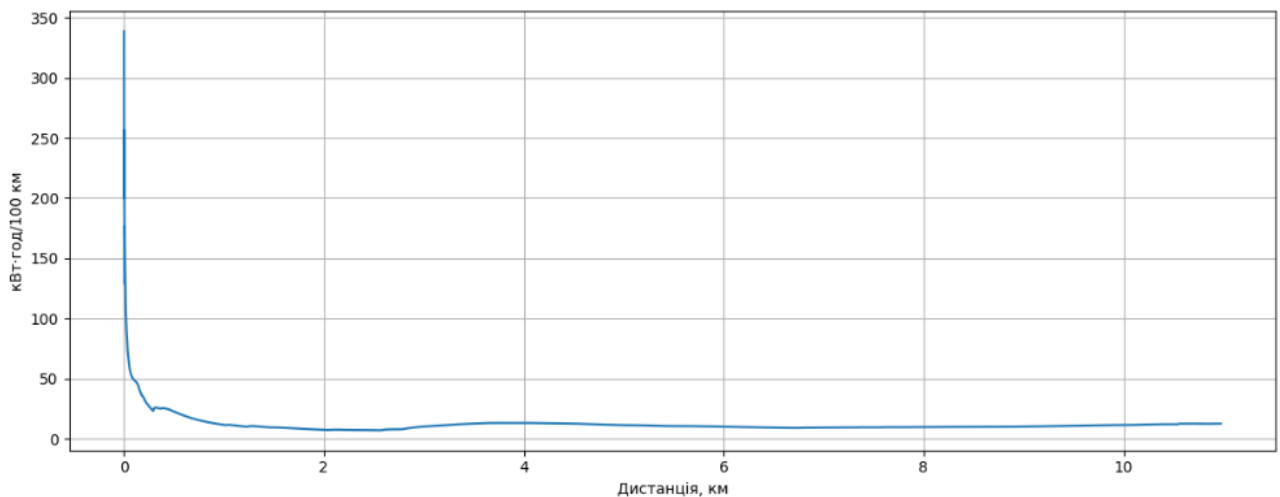


Рисунок 3.7 – Графік питомого енергоспоживання електромобіля

Додатково було проаналізовано розподіл енергоспоживання за режимами руху, представлений на Рис. 3.8. Найбільша частка витраченої енергії припадає на міський та приміський режим експлуатації, що узгоджується з характером руху транспортного засобу під час тестування.

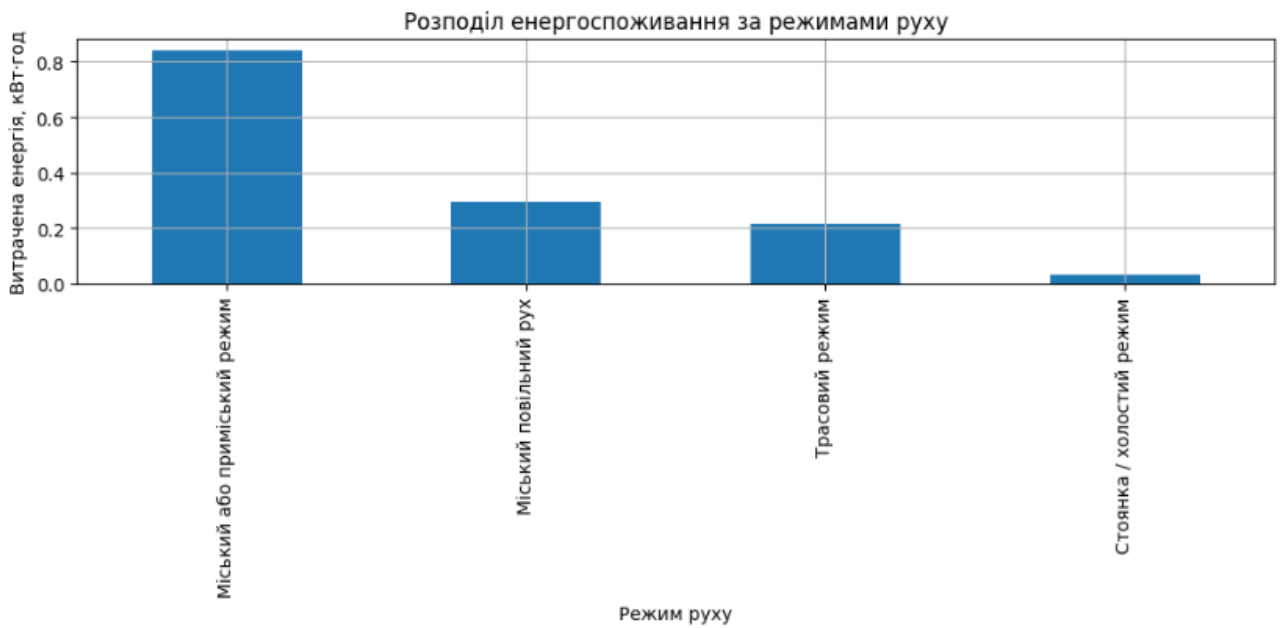


Рисунок 3.8 – Розподіл енергоспоживання за режимами руху

Отримані результати підтверджують правильність роботи алгоритмів аналізу експлуатаційних даних та енергетичних розрахунків.

Перевірка стабільності роботи програмної системи проводилася шляхом багаторазового запуску алгоритмів обробки даних із різним набором вхідних файлів. Під час тестування система коректно функціонувала як у режимі повного аналізу з використанням OBD-, CALC- та батарейних файлів, так і в режимі часткового аналізу при відсутності окремих джерел даних.

Важливою особливістю розробленої системи є наявність механізму автоматичної обробки помилок. Якщо файл має некоректне кодування або відсутні окремі параметри, програма не завершує роботу аварійно, а переходить до обробки доступної інформації. Такий підхід значно підвищує надійність функціонування системи в реальних умовах експлуатації.

Особлива увага приділялася стабільності роботи батарейного моніторингу. Аналіз графіка розбалансу комірок показав, що більшу частину часу різниця напруг між комірками перебувала в межах допустимих значень. Але, в певні моменти часу спостерігалися короткочасні піки до 0,24 В, які були автоматично виявлені алгоритмом контролю.

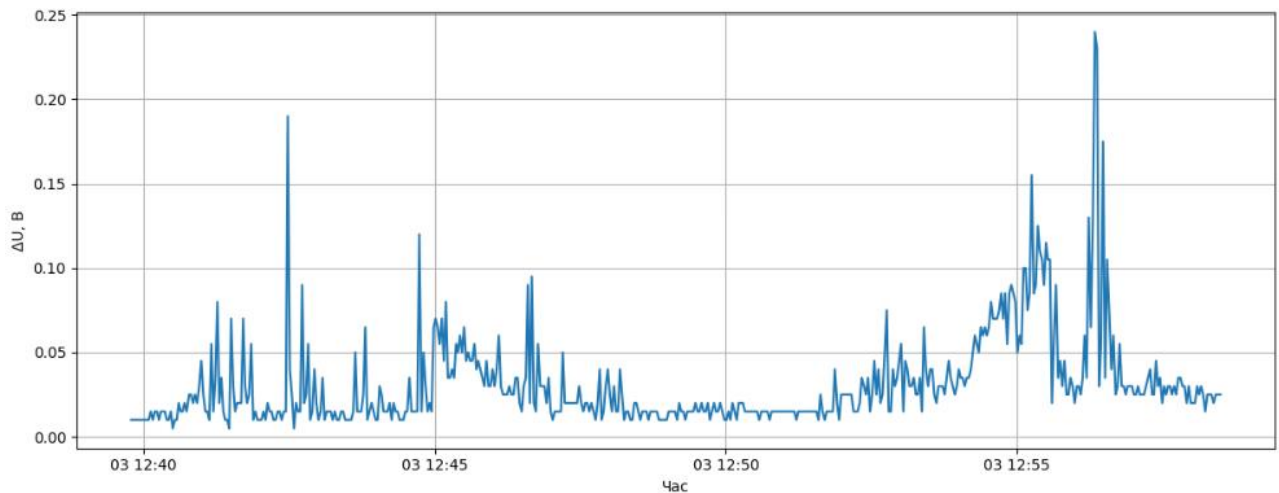


Рисунок 3.9 – Графік розбалансу напруг батареїних комірок

Проведене тестування підтвердило стабільність функціонування програмної системи, коректність реалізації алгоритмів аналізу енергоспоживання та працездатність механізмів автоматичного виявлення аномалій. Результати експериментів показали, що розроблене програмне забезпечення може використовуватися як інструмент моніторингу експлуатаційних характеристик електромобілів, оцінювання енергетичної ефективності та контролю технічного стану тягової батареї.

3.2. Аналіз результатів моніторингу енергоспоживання

Аналіз результатів моніторингу проводився на основі експлуатаційних даних електромобіля, отриманих під час тестової поїздки тривалістю 19,33 хвилини. За цей період програмна система обробила 538 записів телеметричної інформації та розрахувала основні показники енергоспоживання, стану батареї та режимів руху. Отримані результати дозволили оцінити ефективність використання електричної енергії, визначити характер зміни навантаження на акумуляторну батарею та виявити закономірності у споживанні енергії залежно від умов експлуатації.

Результати моніторингу показали, що електромобіль експлуатувався у змішаному режимі руху, який включав міські, приміські та трасові ділянки.

Середня швидкість за весь період спостереження становила 34,89 км/год, тоді як максимальна швидкість досягала 86 км/год. Загальна розрахована дистанція склала 10,97 км.

Середнє енергоспоживання за результатами обробки даних становило 12,59 кВт·год/100 км, що відповідає типовим значенням сучасних електромобілів компактного та середнього класу. Аналіз розподілу витраченої енергії показав, що найбільша частка енергії була використана під час міського та приміського режиму руху. Це пов'язано з частими прискореннями та зміною швидкості, які збільшують навантаження на тяговий електропривод.

У трасовому режимі спостерігалось більш стабільне навантаження на батарею, однак зростання швидкості призводило до збільшення миттєвої потужності. Під час стоянки та руху на малих швидкостях витрати енергії були мінімальними та визначалися переважно роботою допоміжних систем електромобіля.

Отримані результати свідчать, що умови експлуатації безпосередньо впливають на рівень енергоспоживання, а найбільш енерговитратними є режими з інтенсивними змінами швидкості та частими прискореннями.

За результатами моніторингу встановлено, що середня напруга тягової батареї становила 336,02 В, а її значення змінювалися у діапазоні від 285,7 В до 357,6 В. Середній струм батареї дорівнював $-10,98$ А, що свідчить про переважання режиму розряду під час руху.

Загальна витрачена енергія за період тестування склала 1,3815 кВт·год, тоді як система рекуперації повернула до батареї 0,2875 кВт·год. Таким чином, частка рекуперованої енергії становила приблизно 20,8 % від загального обсягу спожитої енергії, що підтверджує ефективну роботу системи рекуперативного гальмування.

Рівень заряду батареї зменшився з 55 % до 45 %, тобто зміна SOC становила 10 %. При цьому температурний режим залишався стабільним: середня температура батареї становила 7,78 °С, мінімальна – 7 °С, а

максимальна – 8,9 °С. Такі значення свідчать про роботу акумуляторної системи у допустимому тепловому діапазоні без ознак перегріву.

Аналіз стану батарейних комірок показав, що максимальний розбаланс напруг досягав 0,24 В. Більшу частину часу значення розбалансу перебували нижче 0,08 В, однак окремі пікові значення були автоматично зафіксовані системою як потенційні аномалії. Це може свідчити про нерівномірність навантаження окремих комірок або початкові прояви їх деградації.

Аналіз часових залежностей показав чіткий взаємозв'язок між швидкістю руху, потужністю батареї та енергоспоживанням. Зі збільшенням швидкості автомобіля спостерігалось зростання миттєвої потужності та інтенсивності розряду акумулятора. Найбільше навантаження на батарею виникало під час прискорень та руху зі швидкістю понад 70 км/год.

Під час гальмування та зниження швидкості система рекуперації забезпечувала повернення частини енергії до батареї, що відображалось появою додатних значень потужності. Максимальна потужність рекуперації досягала 20,72 кВт, тоді як максимальна потужність розряду становила 37,39 кВт.

Графік питомого енергоспоживання показав наявність короточасних пікових значень на початку руху, після чого показник стабілізувався на рівні 10–15 кВт·год/100 км. Це свідчить про адаптацію системи після початкового етапу руху та встановлення сталого режиму роботи електроприводу.

Встановлено також, що збільшення розбалансу напруг комірок відбувалось переважно у періоди підвищеного навантаження на батарею. Загалом, результати моніторингу підтверджують наявність залежності між режимом експлуатації електромобіля, навантаженням на акумуляторну батарею та рівнем енергоспоживання електричної енергії.

3.3. Перспективи вдосконалення системи

Розроблена програмна система моніторингу та аналізу енергоспоживання електромобіля успішно реалізує функції завантаження експлуатаційних даних, їх

обробки, розрахунку енергетичних показників, виявлення аномалій та формування аналітичних звітів. Разом із тим сучасний розвиток електромобільного транспорту, телематичних технологій та інтелектуальних систем керування створює передумови для подальшого вдосконалення програмного забезпечення та розширення його функціональних можливостей.

Подальший розвиток системи може бути спрямований на збільшення кількості підтримуваних форматів даних, реалізацію обробки інформації в режимі реального часу, інтеграцію з мобільними платформами та використання методів штучного інтелекту для прогнозування енергоспоживання й технічного стану акумуляторної батареї. Це дозволить перетворити розроблене програмне забезпечення з локальної аналітичної системи на повноцінний інструмент моніторингу електромобілів.

Одним із напрямків удосконалення системи є розширення кількості підтримуваних джерел експлуатаційних даних. В поточній реалізації програма працює з TXT-файлами типів PID, OBD, CALC та даними батарейних модулів. В подальшому, доцільно реалізувати підтримку форматів CSV, JSON та прямих логів CAN-шини, що дозволить використовувати систему з різними моделями електромобілів і діагностичних комплексів.

Перспективним напрямком є реалізація безпосереднього підключення до OBD-II адаптерів через Bluetooth або Wi-Fi. У такому випадку програма зможе отримувати дані безпосередньо від транспортного засобу та виконувати аналіз у режимі реального часу без необхідності попереднього збереження лог-файлів.

Також, доцільно розширити перелік аналітичних функцій системи. Зокрема, можна реалізувати автоматичний розрахунок деградації батареї, визначення внутрішнього опору комірок, оцінювання залишкового ресурсу акумулятора та аналіз ефективності рекуперативного гальмування. Додатково програму можна доповнити механізмом автоматичного порівняння декількох поїздок, що дозволить оцінювати зміни характеристик електромобіля у часі.

Перспективним є й впровадження системи повідомлень про критичні події. Наприклад, при перевищенні допустимої температури батареї, значному

розбалансі комірок або зниженні напруги нижче встановленого рівня система зможе автоматично генерувати попередження для користувача.

Подальший розвиток системи пов'язаний з інтеграцією програмного забезпечення із мобільними платформами та хмарною інфраструктурою. Такий підхід дозволить організувати централізоване зберігання та аналіз експлуатаційних даних незалежно від місця перебування користувача.

Інтеграція з мобільними додатками на базі Android або iOS надасть можливість відображати поточний стан батареї, рівень заряду, температуру та енергоспоживання безпосередньо на смартфоні. Користувач зможе переглядати статистику поїздок, отримувати повідомлення про аномальні режими роботи та контролювати технічний стан автомобіля у режимі реального часу.

Використання хмарних сервісів дозволить накопичувати історичні дані за тривалий період експлуатації. Зберігання інформації у хмарі забезпечить резервування даних, доступ із різних пристроїв та можливість централізованого аналізу великих масивів телеметрії. Це особливо актуально для корпоративних автопарків та сервісних центрів, які обслуговують значну кількість електромобілів.

Інтеграція з веб-інтерфейсом, також, дозволить реалізувати інтерактивні панелі моніторингу, де користувачі зможуть переглядати графіки енергоспоживання, карти маршрутів, статистику рекуперації та історію стану акумуляторної батареї. Такий підхід значно підвищить інформативність та зручність використання системи.

Окремим напрямком розвитку є використання API для обміну даними з іншими інформаційними системами. Це дозволить інтегрувати програму із сервісами зарядних станцій, системами диспетчеризації транспорту та платформами управління електромобільною інфраструктурою.

Одним із найбільш перспективних напрямків розвитку є впровадження методів інтелектуального аналізу даних та машинного навчання. Накопичення великого обсягу експлуатаційної інформації створює умови для побудови

прогнозних моделей, здатних автоматично оцінювати технічний стан електромобіля та прогнозувати його подальшу роботу.

На основі історичних даних про швидкість, температуру батареї, напругу комірок, струм навантаження та рівень заряду можуть бути створені моделі прогнозування залишкового запасу ходу. Такі алгоритми дозволять враховувати не лише поточний SOC, а й стиль водіння, температуру навколишнього середовища, профіль маршруту та попередні режими експлуатації.

Інтелектуальний аналіз, також, може використовуватися для прогнозування деградації батареї. Аналіз зміни внутрішніх параметрів акумулятора протягом тривалого часу дозволяє оцінювати швидкість старіння комірок та визначати момент, коли їхні характеристики починають погіршуватися. Це дає можливість своєчасно планувати технічне обслуговування та зменшувати ризик відмови батарейного блока.

Перспективним є застосування алгоритмів автоматичного виявлення аномалій. На відміну від порогових методів, реалізованих у поточній версії програми, інтелектуальні алгоритми здатні виявляти складні приховані залежності між параметрами системи. Наприклад, одночасна зміна температури, напруги та струму може бути автоматично розпізнана як ознака потенційної несправності ще до появи критичних симптомів.

В майбутньому такі технології можуть стати основою систем предиктивної діагностики електромобілів, які не лише реєструють поточний стан транспортного засобу, а й прогнозують його поведінку на основі накопичених експлуатаційних даних. Це дозволить підвищити енергоефективність, безпеку та надійність експлуатації електромобільного транспорту.

Висновки до третього розділу

У третьому розділі проведено тестування розробленої програмної системи на реальних експлуатаційних даних електромобіля та виконано аналіз отриманих результатів. Під час тестування було підтверджено коректність роботи

алгоритмів обробки даних, розрахунку енергетичних показників та виявлення аномальних режимів роботи акумуляторної батареї.

За результатами моніторингу визначено основні характеристики енергоспоживання електромобіля, проаналізовано ефективність використання акумуляторної батареї та встановлено закономірності зміни витрат електроенергії залежно від режимів експлуатації. Виявлено, що найбільший вплив на енергоспоживання мають швидкість руху та навантаження на електропривод, а система рекуперації забезпечує повернення частини енергії до батареї.

Також, визначено перспективи подальшого розвитку програмної системи, які пов'язані з інтеграцією мобільних та хмарних сервісів, розширенням функціональних можливостей програмного забезпечення та впровадженням методів інтелектуального аналізу даних для прогнозування технічного стану електромобілів.

РОЗДІЛ 4. ОХОРОНА ПРАЦІ

4.1. Організаційно-правові основи забезпечення безпеки праці

Під час виконання робіт, пов'язаних із моніторингом та аналізом експлуатаційних даних електромобілів, безпечні умови праці повинні забезпечуватися відповідно до чинного законодавства України у сфері охорони праці. Організація робочого місця аналітика даних, експлуатація комп'ютерної техніки, електронного обладнання та програмно-апаратних засобів повинні відповідати встановленим нормативним вимогам щодо безпеки праці, електробезпеки, пожежної безпеки та виробничої санітарії.

Основним нормативно-правовим документом є Закон України «Про охорону праці», який визначає порядок організації безпечних умов праці, відповідальність роботодавця за створення безпечного виробничого середовища та вимоги до управління охороною праці на підприємстві. Додатково питання безпеки праці регулюються Кодексом законів про працю України, Законом України «Про загальнообов'язкове державне соціальне страхування», Правилами пожежної безпеки в Україні та іншими нормативними документами.

Для робочих місць, обладнаних персональними комп'ютерами, особливого значення набувають нормативи щодо організації робочого простору, параметрів освітлення, мікроклімату приміщень, допустимих рівнів шуму та електромагнітного випромінювання. Робота з комп'ютерною технікою повинна виконуватися із дотриманням регламентованих режимів праці та відпочинку, що дозволяє зменшити негативний вплив тривалого використання моніторів на зорову систему та опорно-руховий апарат працівників.

Під час експлуатації комп'ютерного обладнання особлива увага приділяється питанням електробезпеки. Усі технічні засоби повинні бути підключені до мережі через справні електричні розетки із захисним заземленням. Забороняється використання пошкоджених кабелів живлення, несправних

джерел безперебійного живлення та електричних подовжувачів, які не відповідають навантаженню підключеного обладнання.

Під час виконання робіт з аналізу експлуатаційних даних електромобілів працівник має право на належні, безпечні та здорові умови праці, використання справного обладнання, отримання необхідної інформації про потенційні небезпеки та забезпечення засобами колективного захисту. Працівник також має право на проходження інструктажів з охорони праці, навчання безпечним методам роботи та отримання інформації про дії у випадку аварійних ситуацій.

Роботодавець зобов'язаний забезпечити функціонування системи управління охороною праці, організувати безпечне робоче місце, здійснювати контроль технічного стану обладнання та підтримувати нормативні параметри виробничого середовища. Особлива увага приділяється справності комп'ютерної техніки, систем електроживлення, мережевого обладнання та засобів захисту інформації.

Працівник, у свою чергу, повинен дотримуватися встановлених правил безпеки, використовувати обладнання відповідно до його призначення, негайно повідомляти про виявлені несправності та не виконувати роботи, які можуть створити загрозу для життя і здоров'я. Перед початком роботи необхідно перевіряти справність комп'ютерного обладнання, стан кабельних з'єднань та відсутність механічних пошкоджень електричних мереж.

Під час роботи з програмним забезпеченням для аналізу даних особливе значення має дотримання режимів праці та відпочинку. Тривале перебування перед монітором без перерв може призводити до підвищеного навантаження на органи зору та нервову систему. Тому роботодавець повинен організувати робочий процес таким чином, щоб забезпечити оптимальне чергування періодів роботи та відпочинку.

Виконання взаємних обов'язків працівника та роботодавця дозволяє забезпечити безпечне функціонування робочого місця та знизити ризик виникнення професійних захворювань і виробничого травматизму.

Роботи з аналізу експлуатаційних даних електромобілів виконуються переважно в офісних приміщеннях із використанням персональних комп'ютерів та спеціалізованого програмного забезпечення. Безпечна організація таких робіт передбачає створення комфортного та безпечного виробничого середовища, яке відповідає санітарно-гігієнічним та технічним вимогам.

Робоче місце повинно бути обладнане персональним комп'ютером, монітором, клавіатурою, маніпулятором типу «миша», джерелом безперебійного живлення та мережевим обладнанням. Розміщення технічних засобів повинно забезпечувати зручний доступ до обладнання та виключати можливість випадкового пошкодження кабелів живлення або мережевих комунікацій.

Монітор розташовується на відстані 60–70 см від очей користувача, при цьому верхня межа екрана повинна знаходитися на рівні очей, або трохи нижче. Робоче крісло повинно мати регулювання висоти сидіння та кута нахилу спинки, що дозволяє підтримувати фізіологічно правильне положення тіла під час роботи.

Під час обробки великих масивів телеметричних даних навантаження на обчислювальну систему може суттєво збільшуватися. Тому, необхідно забезпечувати ефективне охолодження комп'ютерного обладнання та контролювати температуру роботи процесора, накопичувачів і блоків живлення. Для захисту від втрати даних доцільно використовувати джерела безперебійного живлення та резервне копіювання результатів аналізу.

Організація безпечних умов праці, також, передбачає підтримання належного стану робочого приміщення, своєчасне технічне обслуговування обладнання та контроль відповідності параметрів виробничого середовища встановленим нормативам.

4.2. Характеристика об'єкта та виявлення потенційних небезпек

Об'єктом дослідження є робоче місце аналітика експлуатаційних даних електромобілів, діяльність якого пов'язана зі збиранням, обробкою, аналізом та візуалізацією телеметричної інформації, отриманої від систем електромобіля. Основними завданнями працівника є аналіз параметрів тягової батареї, оцінювання енергоспоживання транспортного засобу, виявлення аномалій у роботі електричних систем, обробка масивів даних та формування технічних звітів.

Робоче місце розташовується в адміністративному або офісному приміщенні та обладнується персональним комп'ютером або робочою станцією. Для виконання розрахунків та аналізу великих обсягів телеметричних даних доцільно використовувати комп'ютер із багатоядерним процесором продуктивністю не менше 3,0 ГГц, оперативною пам'яттю обсягом від 16 ГБ та твердотільним накопичувачем SSD обсягом не менше 512 ГБ. Використання SSD-накопичувачів забезпечує високу швидкість обробки великих наборів експлуатаційних даних та скорочує час формування аналітичних звітів.

До складу робочого місця входять монітор діагоналлю 24–27 дюймів із роздільною здатністю не менше Full HD (1920×1080), клавіатура, маніпулятор типу «миша», джерело безперебійного живлення, маршрутизатор локальної мережі та засоби резервного копіювання інформації. Для роботи з графіками та таблицями доцільним є використання двох моніторів, що підвищує ефективність аналізу даних та зменшує кількість перемикань між програмними вікнами.

У процесі роботи використовуються програмні засоби Python, Google Colab, Jupyter Notebook, Pandas, NumPy, Matplotlib, OpenPyXL та офісні програмні пакети для оформлення звітної документації. Під час виконання обчислень процесор та оперативна пам'ять працюють у режимі підвищеного навантаження, що супроводжується збільшенням тепловиділення комп'ютерної техніки та інтенсивною роботою систем охолодження.

Робоча зона повинна забезпечувати площу не менше 6 м^2 на одного працівника при об'ємі приміщення не менше 20 м^3 . Висота приміщення повинна становити не менше 3 м. Відстань між бічними поверхнями моніторів рекомендується не менше 1,2 м, а між тильною поверхнею одного монітора та екраном іншого – не менше 2,5 м. Такі параметри забезпечують комфортні умови праці та знижують вплив електромагнітних полів.

Робочий день аналітика даних переважно пов'язаний із тривалим перебуванням за комп'ютером. Тривалість безперервної роботи з монітором може становити від 2 до 4 годин під час аналізу великих масивів телеметричних даних. Тому, особливого значення набувають ергономічні параметри робочого місця, правильне розташування обладнання та дотримання режимів праці та відпочинку.

Під час виконання робіт з аналізу експлуатаційних даних електромобілів основними небезпечними факторами є електричний струм, що використовується для живлення комп'ютерного обладнання, мережевих пристроїв та допоміжної техніки. Напруга живлення персонального комп'ютера становить 220 В змінного струму, що у разі пошкодження ізоляції або порушення цілісності електричних мереж може створювати небезпеку ураження працівника електричним струмом. Особливо небезпечними є несправні блоки живлення, пошкоджені кабелі, відсутність заземлення та перевантаження електричних подовжувачів.

Важливим фактором є мікроклімат виробничого приміщення. Значна кількість комп'ютерної техніки створює додаткове теплове навантаження на приміщення. Один сучасний системний блок під час виконання ресурсоємних обчислень може виділяти від 250 до 500 Вт теплової енергії. У разі одночасної роботи декількох робочих станцій температура повітря в приміщенні може швидко підвищуватися. Для забезпечення нормальних умов праці температура повітря повинна підтримуватися в межах $20\text{--}24 \text{ }^\circ\text{C}$ у холодний період року та $22\text{--}25 \text{ }^\circ\text{C}$ у теплий період року. Відносна вологість повітря повинна становити $40\text{--}60 \%$, а швидкість руху повітря не перевищувати $0,1\text{--}0,2 \text{ м/с}$.

Освітлення робочого місця має суттєвий вплив на продуктивність праці та стан зорового апарату працівника. Під час роботи з таблицями, графіками та програмним кодом необхідно забезпечити освітленість робочої поверхні на рівні 300–500 лк. Недостатня освітленість призводить до швидкої втоми очей, а надмірна яскравість викликає появу відблисків на екрані монітора. Коефіцієнт природної освітленості для таких приміщень повинен становити не менше 1,5 %.

Джерелом шуму є вентилятори процесорів, систем охолодження відеокарт, блоків живлення, серверного обладнання та систем кондиціонування. Під час звичайної роботи рівень шуму становить 35–45 дБА, а під час виконання складних розрахунків може досягати 50 дБА. Хоча ці значення не перевищують допустимих нормативів для офісних приміщень, тривалий вплив шуму може викликати втому та зниження концентрації уваги.

У процесі експлуатації комп'ютерного обладнання утворюються електромагнітні поля низької інтенсивності. Основними джерелами електромагнітного випромінювання є монітори, блоки живлення, маршрутизатори, мережеве обладнання та джерела безперебійного живлення. Сучасні пристрої відповідають міжнародним стандартам електромагнітної сумісності, однак при великій кількості обладнання в одному приміщенні сумарний вплив електромагнітних полів збільшується.

Найбільш характерним шкідливим фактором для даної професійної діяльності є психофізіологічне навантаження. Аналіз телеметричних даних потребує постійної концентрації уваги, роботи з великими масивами числової інформації, перевірки графіків, таблиць та результатів обчислень. Тривале статичне положення тіла призводить до навантаження на шийний відділ хребта, плечовий пояс та поперековий відділ. Крім того, тривала робота за монітором викликає зорову втому через постійне фокусування погляду на екрані.

Найбільш імовірною аварійною ситуацією на робочому місці аналітика експлуатаційних даних є виникнення пожежі внаслідок короткого замикання електричної мережі або перегріву електрообладнання. Основними джерелами займання можуть бути блоки живлення комп'ютерів, джерела безперебійного

живлення, перевантажені подовжувачі, пошкоджені кабельні лінії та несправне мережеве обладнання. Особливу небезпеку становить накопичення пилу всередині системних блоків, що погіршує тепловідведення та підвищує ризик перегріву електронних компонентів.

До аварійних ситуацій належить раптове відключення електроживлення. У такому випадку можливе пошкодження файлів, втрата результатів розрахунків та порушення роботи програмного забезпечення. Для забезпечення безперервності роботи необхідно використовувати джерела безперебійного живлення з автономністю не менше 10–15 хвилин, що дозволяє коректно завершити роботу програм та зберегти результати аналізу.

Іншою потенційною небезпекою є відмова комп'ютерного обладнання під час виконання ресурсоємних обчислень. Перегрів процесора, оперативної пам'яті чи накопичувачів може призвести до аварійного завершення програм, втрати частини даних або виходу обладнання з ладу. Для мінімізації ризику необхідно регулярно контролювати температурний режим компонентів та проводити технічне обслуговування систем охолодження.

В сучасних умовах значну увагу необхідно приділяти небезпекам воєнного характеру. Під час повітряної тривоги працівники повинні негайно припинити виконання робіт, відключити електрообладнання від мережі та переміститися до захисної споруди або укриття. Особливої актуальності набуває захист інформаційних ресурсів від втрати внаслідок аварійних ситуацій, тому резервне копіювання даних повинно виконуватися регулярно.

До загальних небезпек, також, належать пожежі в будівлі, задимлення приміщення, руйнування інженерних мереж, затоплення, техногенні аварії та інші надзвичайні ситуації природного, або антропогенного характеру. Для забезпечення безпеки працівників приміщення повинно бути обладнане системою пожежної сигналізації, вогнегасниками, евакуаційним освітленням та схемами евакуації.

Проведений аналіз показує, що для робочого місця аналітика експлуатаційних даних електромобілів найбільш значущими є ризики, пов'язані

з електробезпекою, пожежною безпекою, відмовою комп'ютерного обладнання, несприятливими параметрами мікроклімату та психофізіологічними навантаженнями.

4.3. Дослідження ризику реалізації небезпек та заходи безпеки

Після проведення аналізу робочого місця аналітика експлуатаційних даних електромобілів та виявлення небезпечних і шкідливих факторів виробничого середовища необхідно виконати оцінювання ризику їх реалізації. Метою оцінювання ризиків є визначення найбільш небезпечних подій, оцінка ймовірності їх виникнення та можливих наслідків для працівників і технічного обладнання.

Для оцінки ризиків використано матричний метод, відповідно до якого рівень ризику визначається як поєднання ймовірності виникнення небезпечної події та тяжкості її наслідків. Для аналізу обрано найбільш характерні небезпеки для робочого місця аналітика експлуатаційних даних електромобілів: ураження електричним струмом, пожежа внаслідок короткого замикання, відмова комп'ютерного обладнання, погіршення умов мікроклімату та психофізіологічне перевантаження працівника.

Для кількісної оцінки ризиків використано п'ятибальну шкалу ймовірності та тяжкості наслідків. Рівень ризику визначається як добуток цих показників.

Таблиця 4.1 – Матриця оцінювання ризиків небезпечних подій

Небезпечна подія	Ймовірність	Тяжкість наслідків	Рівень ризику
1	2	3	4
Ураження електричним струмом	2	5	10
Пожежа через коротке замикання	2	5	10

Продовження таблиці 4.1

1	2	3	4
Відмова комп'ютерного обладнання	4	2	8
Погіршення параметрів мікроклімату	3	2	6
Психофізіологічне перевантаження	4	3	12

Отримані результати показують, що найбільший рівень ризику для даного робочого місця має психофізіологічне перевантаження працівника. Це пов'язано з тривалою роботою за комп'ютером, необхідністю аналізу великих обсягів інформації, високою концентрацією уваги та виконанням однотипних операцій протягом тривалого часу.

Значним також є ризик ураження електричним струмом та виникнення пожежі. Незважаючи на відносно невисоку ймовірність таких подій, тяжкість можливих наслідків є максимальною, оскільки вони можуть призвести до травмування персоналу, виходу обладнання з ладу та припинення роботи підприємства.

Ризик відмови комп'ютерного обладнання оцінюється як середній. Втрата працездатності робочої станції не створює безпосередньої загрози життю працівника, однак може призвести до втрати результатів аналізу, простою та фінансових збитків.

Погіршення параметрів мікроклімату має відносно невисокий рівень ризику, проте при тривалому впливі може призводити до зниження працездатності персоналу та збільшення кількості помилок під час виконання аналітичних робіт.

Для зменшення ризику ураження електричним струмом необхідно використовувати тільки справне комп'ютерне обладнання, підключене до мережі через розетки із захисним заземленням. Електричні кабелі повинні бути ізольованими та захищеними від механічних пошкоджень. Забороняється

експлуатація несправних подовжувачів, блоків живлення та джерел безперебійного живлення.

Для зниження пожежної небезпеки необхідно забезпечити регулярний контроль технічного стану електрообладнання, очищення системних блоків від пилу та перевірку температурного режиму роботи компонентів. У приміщенні повинні бути встановлені порошкові або вуглекислотні вогнегасники, а також засоби пожежної сигналізації.

Для мінімізації ризику втрати даних унаслідок відмови обладнання необхідно використовувати джерела безперебійного живлення потужністю не менше 800–1000 ВА, виконувати регулярне резервне копіювання інформації та застосовувати сучасні SSD-накопичувачі з високою надійністю зберігання даних.

Зниження ризиків, пов'язаних із мікрокліматом приміщення, забезпечується використанням систем кондиціонування та вентиляції. Температура повітря повинна підтримуватися в межах нормативних значень, а робочі місця повинні розташовуватися таким чином, щоб виключити вплив прямих сонячних променів на екрани моніторів.

Для боротьби з психофізіологічним перевантаженням необхідно дотримуватися раціонального режиму праці та відпочинку. Під час виконання аналітичних робіт рекомендується робити короткочасні перерви для зниження навантаження на зоровий апарат та опорно-рухову систему. Доцільним є використання ергономічних крісел, регульованих столів та моніторів із технологіями захисту зору.

Застосування наведених заходів дає змогу суттєво знизити рівень професійних ризиків та підвищити безпеку виконання робіт з аналізу експлуатаційних даних електромобілів.

Для забезпечення безпечних умов праці робоче місце аналітика повинно відповідати ергономічним та санітарно-гігієнічним вимогам. Відстань від очей користувача до екрана монітора повинна становити 60–70 см, а верхня межа екрана має знаходитися на рівні очей або трохи нижче. Висота робочого столу

повинна перебувати в межах 720–750 мм, а крісло повинно забезпечувати можливість регулювання висоти сидіння та кута нахилу спинки.

Освітленість робочої поверхні необхідно підтримувати на рівні 300–500 лк. Для зменшення втоми очей рекомендується використовувати монітори з антивідблисковим покриттям та технологіями зниження інтенсивності синього спектра випромінювання.

Під час роботи з комп'ютерною технікою необхідно забезпечувати регулярне провітрювання приміщення та підтримання температури повітря у межах 20–24 °С. Відносна вологість повітря повинна становити 40–60 %, що забезпечує комфортні умови праці та нормальну роботу електронного обладнання.

Для забезпечення електробезпеки всі електричні установки повинні мати захисне заземлення та автоматичні вимикачі захисту. Забороняється виконувати будь-які ремонтні роботи електрообладнання без відключення його від мережі живлення.

З метою пожежної безпеки у приміщенні повинні бути встановлені первинні засоби пожежогасіння, евакуаційні схеми та засоби оповіщення про пожежу. Працівники повинні проходити інструктаж щодо дій у разі виникнення пожежі або іншої надзвичайної ситуації.

Особлива увага повинна приділятися захисту інформації та резервному копіюванню результатів аналізу. Наявність резервних копій дозволяє забезпечити безперервність роботи, навіть, у випадку аварійного виходу обладнання з ладу, або втрати електроживлення.

Висновки до четвертого розділу

У розділі виконано аналіз організаційно-правових аспектів охорони праці, проведено характеристику робочого місця аналітика експлуатаційних даних електромобілів та визначено основні небезпечні й шкідливі виробничі фактори. Встановлено, що найбільший вплив на безпеку праці мають електротехнічні

ризиками, пожежна безпека, психофізіологічне навантаження працівника, можливі відмови комп'ютерного обладнання та несприятливі параметри мікроклімату.

Проведено оцінювання ризиків найбільш характерних небезпечних подій із використанням матричного методу та визначено рівні їх безпеки. За результатами аналізу розроблено комплекс організаційних і технічних заходів щодо зниження професійних ризиків, підвищення електричної та пожежної безпеки, забезпечення належних умов праці та збереження працездатності персоналу.

Реалізація запропонованих заходів дозволяє забезпечити безпечне виконання робіт з моніторингу та аналізу експлуатаційних даних електромобілів, знизити ймовірність виникнення аварійних ситуацій та створити комфортні умови для ефективної професійної діяльності працівників.

ВИСНОВКИ

У кваліфікаційній роботі розв'язано актуальну задачу розробки програмної системи моніторингу та аналізу енергоспоживання електромобіля на основі експлуатаційних даних. Актуальність дослідження обумовлена стрімким розвитком електромобільного транспорту, необхідністю підвищення енергоефективності транспортних засобів та потребою у створенні інструментів аналізу технічного стану акумуляторних систем під час реальної експлуатації.

В процесі виконання роботи проведено аналіз особливостей енергоспоживання електромобілів та визначено основні фактори, що впливають на витрати електричної енергії. Встановлено, що найбільший вплив на рівень енергоспоживання здійснюють швидкість руху, стиль керування транспортним засобом, режими прискорення та гальмування, температурний режим експлуатації, навантаження на електропривод та ефективність роботи системи рекуперації. Проаналізовано структуру експлуатаційних даних електромобіля та визначено параметри, необхідні для побудови системи моніторингу.

У роботі досліджено сучасні методи збору та аналізу телеметричних даних транспортних засобів. Розглянуто використання OBD-інтерфейсів, CAN-шини та систем керування батареєю BMS як основних джерел інформації про стан електромобіля. Визначено доцільність використання часових рядів для синхронізації параметрів та побудови інтегрованого набору експлуатаційних даних.

На основі проведеного аналізу обґрунтовано структуру програмної системи моніторингу енергоспоживання. Запропонована архітектура побудована за модульним принципом і включає підсистеми завантаження даних, автоматичного визначення типу файлів, розшифрування CAN/PID-повідомлень, обробки OBD-параметрів, аналізу стану батареї, формування аналітичних звітів та графічної візуалізації результатів.

У межах практичної частини розроблено програмне забезпечення мовою Python із використанням бібліотек Pandas, NumPy, Matplotlib та OpenPyXL.

Реалізовано механізм автоматичного завантаження та обробки TXT-файлів із різною структурою даних. Програма забезпечує підтримку файлів типів PID INT, PID HEX, OBD, CALC, CELLS, CELL TEMPERATURES та INFO LOG, що дозволяє виконувати комплексний аналіз експлуатаційних параметрів електромобіля.

Розроблено алгоритм автоматичного розшифрування CAN/PID-даних із перетворенням байтової інформації у числові параметри та ASCII-представлення. Реалізовано механізм формування єдиної таблиці експлуатаційних даних шляхом об'єднання інформації з різних джерел за часовими мітками. Це дозволило створити інтегрований набір даних для подальшого енергетичного аналізу.

У програмній системі реалізовано розрахунок основних енергетичних показників електромобіля, зокрема миттєвої потужності батареї, витраченої та рекуперованої енергії, пройденої дистанції, питомого енергоспоживання та режимів руху транспортного засобу. Також реалізовано алгоритм класифікації режимів експлуатації на стоянку, міський, приміський, трасовий та високошвидкісний режими руху.

Окрему увагу приділено аналізу технічного стану тягової батареї. У програмі реалізовано контроль напруг комірок, температур батарейних модулів, рівня заряду SOC та розбалансу акумуляторних елементів. Додатково впроваджено систему автоматичного виявлення аномалій, яка дозволяє реєструвати перевищення допустимих меж напруги, температури та нерівномірності роботи комірок батареї.

Проведене тестування програмної системи на реальних експлуатаційних даних підтвердило її працездатність та коректність реалізованих алгоритмів. У ході експериментів було оброблено 538 записів телеметричної інформації за період 19,33 хвилини експлуатації електромобіля. Встановлено, що середня швидкість руху становила 34,89 км/год, максимальна швидкість досягала 86 км/год, а загальна дистанція склала 10,97 км.

За результатами аналізу визначено, що середнє енергоспоживання електромобіля становило 12,59 кВт·год/100 км, а загальна витрачена енергія склала 1,3815 кВт·год. Система рекуперації забезпечила повернення 0,2875 кВт·год енергії до акумуляторної батареї, що становить близько 20,8 % від загального обсягу спожитої енергії. Це свідчить про ефективну роботу системи рекуперативного гальмування.

Під час тестування встановлено, що рівень заряду батареї зменшився з 55 % до 45 %, а температурний режим залишався стабільним у межах від 7 °С до 8,9 °С, що відповідає нормальним умовам експлуатації. Максимальний розбаланс напруг комірок досягав 0,24 В, що було автоматично зафіксовано системою як потенційна аномалія.

Аналіз результатів моніторингу показав наявність прямої залежності між швидкістю руху, навантаженням на батарею та рівнем енергоспоживання. Встановлено, що найбільші витрати енергії спостерігаються у режимах інтенсивного прискорення та руху на підвищених швидкостях, тоді як рекуперативне гальмування дозволяє частково компенсувати втрати електроенергії.

Розглянуто перспективи подальшого розвитку системи, які передбачають інтеграцію з OBD-II адаптерами в режимі реального часу, використання мобільних та хмарних сервісів, а також впровадження методів машинного навчання для прогнозування запасу ходу, деградації батареї та автоматичного виявлення несправностей.

Загалом, поставлені у роботі завдання повністю виконані, а мета дослідження досягнута. Розроблена програмна система є працездатним програмним засобом моніторингу та аналізу енергоспоживання електромобіля, який може використовуватися для оцінювання енергоефективності транспортного засобу, контролю технічного стану акумуляторної батареї та подальшого розвитку інтелектуальних систем керування електромобілями.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Безпека в надзвичайних ситуаціях : навч. посібник для студентів ЗВО України : у 2 ч. Ч. 1: Надзвичайні ситуації / М. Л. Лисиченко, В. В. Вамболь, С. О. Вамболь, М. М. Кірієнко, І. А. Черепньов, В. М. Власовець ; за ред. М. Л. Лисиченка ; ХНТУСГ. – Харків : ТОВ “ПромАрт”, 2021. – 202 с.
2. Бородкіна І. Інженерія програмного забезпечення: навч. посібник. - К.: Центр учбової літератури, 2021. - 204 с.
3. Джеймс Л., Грін Б. Класифікація і проектування електромобілів. – Лондон: Routledge, 2019. – 450 с.
4. Енергетична ефективність електромобілів: аналітичне дослідження / С.В. Андрєєв, О.Г. Павленко. – Одеса: Наука і техніка, 2020. – 280 с.
5. Іванов А.В. Гібридні транспортні засоби: Навчальний посібник. Харків: ХНАДУ, 2022. – 210 с.
6. Костюк С.М., Петренко О.В. Прогнозування енергоспоживання з використанням хмарних технологій. Вісник енергетики, 2020, 45(2): с. 52-63.
7. Красовський Ю.Л., Кулик В.В., Лежнюк П.Д. Керування втратами електроенергії в розподільних мережах з використанням засобів АСКОЕ. Вісник Харківського держ. техн. ун-ту сільського господ. 2003. Вип. 19. Т.1. С. 99–107.
8. Методики оцінки енергетичних і екологічних параметрів автомобілів / Під ред. Гречанюка О.В. – Київ: Транспорт і екологія, 2021. – 360 с.
9. Міронов Д.В. Електричні та гібридні транспортні засоби. Методичні вказівки до виконання практичних робіт / Д.В. Міронов, Тесля В.О. – Тернопіль: ТНТУ, 2024. – 65 с.
10. Міронов Д.В. Електронне та мікропроцесорне обладнання автомобілів: Методичні вказівки до виконання лабораторних робіт / Д.В. Міронов – Тернопіль: ТНТУ, 2023. – 62 с.

11. Міронов Д.В. Конспект лекцій з дисципліни «Електричні та гібридні транспортні засоби» для студентів спеціальності 274 «Автомобільний транспорт» денної та заочної форми навчання / Укл.: Д.В. Міронов – Тернопіль: ТНТУ, 2024. – 154 с.
12. Охорона праці в галузі та цивільний захист: навчальний посібник / Ю. А. Гасило, О. А. Крюковська. К. О. Левчук, Р. Я. Романюк. — Кам'янське : ДДТУ, 2017. — 369 с.
13. Охорона праці на автомобільному транспорті : навчальний посібник / Пістун І. П., Хом'як Й. В., Хом'як В. В. - 2-ге вид., стер. - Суми : Університетська книга, 2015. - 374 с.
14. Петров М.В. Основи електротехніки для транспортних засобів. – Харків: Техніка і транспорт, 2018. – 372 с.
15. Розрахунки енергоефективності транспортних систем / Під ред. Ковалю В.М. – Львів: Національний університет «Львівська політехніка», 2019. – 300 с.
16. Рябчиков, Д. М. Дослідження енергоспоживання електромобілів та автономних систем живлення в транспортній інфраструктурі. Ч. 1 : Визначення ефективності фотоелектричних панелей на даху електромобілів для підвищення дальності пробігу : дипломна робота ... бакалавра : 141 Електроенергетика, електротехніка та електромеханіка / Рябчиков Данило Миколайович. – Харків : ХНАДУ, 2025. – 74 с.
17. Скрипник С.О., Шеїна Г.О. Порівняння технологічних втрат електричної мережі 6(10) кВ та 20 кВ. Наукові праці ДонНТУ. Серія: «Електротехніка і енергетика». 2020. №1(22). С. 21–26.
18. Сучасні методи оптимізації енергоспоживання у транспорті / Баранов В.В., Савченко І.О. – Харків: Видавництво ХНАДУ, 2020. – 290 с.
19. Техногенні ризики і транспорт: Монографія / Кравець С.М., Поліщук Г.С. – Львів: ЛНТУ, 2022. – 310 с.

20. Циценков Д.В., Красовський П.Ю. Методи та засоби зниження технічних втрат електроенергії в елементах систем електропостачання. Електротехніка та електроенергетика. 2015. № 1. С. 77–82.
21. Amir, M., Zaheeruddin & Haque, A. (2021). Optimal Scheduling of Charging/Discharging Power and EVs Pattern Using Stochastic Techniques in V2G System. 2021 IEEE Transportation Electrification Conference (ITEC-India), New Delhi, India, 2021, pp. 1-6.
22. Chaturvedi A., Prasad K., Ranjan R. Use of interval arithmetic to incorporate the uncertainty of load demand for radial distribution system analysis, IEEE transactions on power delivery, 2006, 21(2), 1019-1021.
23. Krings, A., & Monissen, C. Review and Trends in Electric Traction Motors for Battery Electric and Hybrid Vehicles. IEEE International Conference on Electrical Machines, 2020, pp. 1–8.
24. Liu, J., Chen, Y., Zhan, J., & Shang, F. (2019). Heuristic dynamic programming based online energy management strategy for plug-in hybrid electric vehicles. IEEE Transactions on Vehicular Technology, 68(5), 4479-4493.
25. Togun, H., Aljibori, H.S.S., Abed, A.M., Biswas, N., Alshamkhani, M.T., Niyas, H., Mohammed, H.I., Rashid, F.L., Dhabab, J.M., & Paul, D. (2024). A review on recent advances on improving fuel economy and performance of a fuel cell hybrid electric vehicle. International Journal of Hydrogen Energy, 89, 22-47.
26. Waldmann, T., Wilka, M., Kasper, M., Fleischhammer, M., and Wohlfahrt-Mehrens, M. (2014). Temperature dependent ageing mechanisms in Lithium-ion batteries – A Post-Mortem study. Journal of Power Sources, 262, 129–135.
27. Wang, Y., Li, Y., Zhang, L. "Cloud-Based Energy Forecasting for Smart Grids." IEEE Transactions on Smart Grid, 2020, 11(2): 1234-1243. Конституція України : Закон України від 28.06.1996 р. № 254к/96-ВР.
28. Про охорону праці : Закон України від 14.10.1992 р. № 2694-ХІІ.
29. Кодекс законів про працю України : Закон України від 10.12.1971 р. № 322-VIII.

30. Кодекс цивільного захисту України : Закон України від 02.10.2012 р. № 5403-VI.
31. Правила пожежної безпеки в Україні : наказ МВС України від 30.12.2014 р. № 1417.
32. ДСанПіН 3.3.2.007-98. Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин. Київ : МОЗ України, 1998.
33. ДБН В.2.5-28:2018. Природне і штучне освітлення. Київ : Мінрегіон України, 2018.
34. ДСН 3.3.6.042-99. Державні санітарні норми мікроклімату виробничих приміщень. Київ : МОЗ України, 1999.
35. ДСТУ EN ISO 12100:2016. Безпечність машин. Загальні принципи конструювання. Оцінювання ризику та зменшення ризику. Київ : ДП «УкрНДНЦ», 2016.
36. ДСТУ ISO 45001:2019. Системи управління охороною здоров'я та безпекою праці. Вимоги та настанови щодо застосування. Київ : ДП «УкрНДНЦ», 2019.
37. Жидецький В. Ц. Основи охорони праці : підручник. 5-те вид., перероб. і допов. Львів : Афіша, 2014. 376 с.
38. Березуцький В. В., Бондаренко Т. С., Васьковець Л. А. Основи охорони праці : навчальний посібник. Харків : ФОП Панов А. М., 2017. 553 с.
39. Гогіташвілі Г. Г., Лапін В. М. Управління охороною праці та ризиком за міжнародними стандартами : навчальний посібник. Львів : ЗУКЦ, 2019. 408 с.
40. Запорожець О. І., Боровик І. М., Русаловський А. В. Безпека життєдіяльності : підручник. Київ : Центр учбової літератури, 2019. 448 с.
41. Безпека життєдіяльності та охорона праці : підручник / О. Г. Левченко, О. І. Полукаров, В. В. Зацарний та ін. Київ : КПІ ім. Ігоря Сікорського, 2019. 420 с.

42. Гігієна праці та виробнича санітарія : підручник / за ред. А. М. Шевченка. Київ : ВСВ «Медицина», 2020. 472 с.
43. Законодавство України про охорону праці : збірник нормативно-правових актів. Київ : Основа, 2021.

ДОДАТКИ

Додаток А

Лістинг програми

```
import os
import re
import io
import zipfile
import warnings
from datetime import datetime

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

warnings.filterwarnings("ignore")

def upload_files():
    """
    Завантаження TXT-файлів у середовищі Colab.
    Якщо код запускається не в Colab, файли можна покласти в поточну папку.
    """
    try:
        from google.colab import files
        uploaded = files.upload()
        paths = []

        for filename, content in uploaded.items():
            with open(filename, "wb") as f:
                f.write(content)
            paths.append(filename)

        return paths

    except Exception:
        print("Colab upload недоступний. Будуть використані TXT-файли з поточної папки.")
        return [f for f in os.listdir(".") if f.lower().endswith(".txt")]

def read_text_file(path):
    """
    Безпечне читання текстового файлу з різними кодуваннями.
    """
    encodings = ["utf-8", "cp1251", "latin1"]

    for enc in encodings:
        try:
```

```

        with open(path, "r", encoding=enc) as f:
            return f.read()
    except UnicodeDecodeError:
        continue

with open(path, "r", errors="ignore") as f:
    return f.read()

def detect_file_type(path, text):
    """
    Автоматичне визначення типу файлу за заголовком і назвою.
    """
    name = os.path.basename(path).lower()
    first_line = text.splitlines()[0].strip() if text.splitlines() else ""

    if first_line.startswith("Time;PID;hex"):
        return "pid_hex"

    if first_line.startswith("Time;PID;int"):
        return "pid_int"

    if first_line.startswith("Time;Parameter;Value"):
        if "calc" in name:
            return "calc"
        if "obd" in name:
            return "obd"
        return "parameter"

    if first_line.startswith("Time;Module;Cell;Volts"):
        return "cells"

    if first_line.startswith("Time;Module;Sensor;Temperature"):
        return "cell_temperatures"

    if "app:" in text[:500] or "obd:" in text[:500].lower():
        return "info_log"

    return "unknown"

def parse_time_column(df):
    """
    Перетворення стовпця Time у формат datetime.
    """
    if "Time" in df.columns:
        df["Time"] = pd.to_datetime(
            df["Time"],
            format="%d-%m-%Y %H:%M:%S.%f",
            errors="coerce"
        )

```

```
)
return df

def safe_numeric(series):
    """
    Безпечно перетворення значень у числа.
    """
    return pd.to_numeric(series, errors="coerce")

def read_semicolon_table(path):
    """
    Читання таблиці з роздільником ';'.
    """
    return pd.read_csv(path, sep=";", engine="python")

def hex_to_int(value):
    try:
        return int(str(value), 16)
    except Exception:
        return np.nan

def bytes_to_u16(b1, b2):
    if pd.isna(b1) or pd.isna(b2):
        return np.nan
    return int(b1) * 256 + int(b2)

def bytes_to_i16(b1, b2):
    value = bytes_to_u16(b1, b2)
    if pd.isna(value):
        return np.nan
    if value >= 32768:
        value -= 65536
    return value

def decode_ascii_from_bytes(byte_values):
    chars = []
    for b in byte_values:
        try:
            b = int(b)
            if 32 <= b <= 126:
                chars.append(chr(b))
        except Exception:
            pass
    return "".join(chars)
```

```

def decode_pid_table(df, source_type):
    """
    Розшифрування сирих CAN/PID-даних.
    Програма не прив'язана до одного PID, а автоматично формує
    набір універсальних технічних полів для кожного повідомлення.
    """
    df = df.copy()
    df = parse_time_column(df)

    if source_type == "pid_hex":
        byte_cols = [c for c in df.columns if c.startswith("hex")]
        for c in byte_cols:
            df[c.replace("hex", "byte")] = df[c].apply(hex_to_int)
        byte_cols = [c.replace("hex", "byte") for c in byte_cols]

    else:
        byte_cols = [c for c in df.columns if c.startswith("int")]
        for c in byte_cols:
            df[c] = safe_numeric(df[c])
        byte_cols = byte_cols

    decoded_rows = []

    for _, row in df.iterrows():
        bytes_list = [row[c] for c in byte_cols if c in row.index]
        bytes_list = [0 if pd.isna(x) else int(x) for x in bytes_list]

        record = {
            "Time": row.get("Time"),
            "PID": str(row.get("PID")),
            "raw_bytes": " ".join([f"{b:02X}" for b in bytes_list]),
            "ascii": decode_ascii_from_bytes(bytes_list),
            "u16_0_1": bytes_to_u16(bytes_list[0], bytes_list[1]) if
len(bytes_list) >= 2 else np.nan,
            "u16_2_3": bytes_to_u16(bytes_list[2], bytes_list[3]) if
len(bytes_list) >= 4 else np.nan,
            "u16_4_5": bytes_to_u16(bytes_list[4], bytes_list[5]) if
len(bytes_list) >= 6 else np.nan,
            "u16_6_7": bytes_to_u16(bytes_list[6], bytes_list[7]) if
len(bytes_list) >= 8 else np.nan,
            "i16_0_1": bytes_to_i16(bytes_list[0], bytes_list[1]) if
len(bytes_list) >= 2 else np.nan,
            "i16_2_3": bytes_to_i16(bytes_list[2], bytes_list[3]) if
len(bytes_list) >= 4 else np.nan,
            "i16_4_5": bytes_to_i16(bytes_list[4], bytes_list[5]) if
len(bytes_list) >= 6 else np.nan,
            "i16_6_7": bytes_to_i16(bytes_list[6], bytes_list[7]) if
len(bytes_list) >= 8 else np.nan,
            "byte_sum": np.sum(bytes_list),
            "byte_mean": np.mean(bytes_list) if bytes_list else np.nan,

```

```

        "source_type": source_type
    }

    decoded_rows.append(record)

decoded = pd.DataFrame(decoded_rows)
return decoded

def pivot_parameter_file(df, prefix=""):
    """
    Перетворення структури Time;Parameter;Value у широку таблицю:
    Time | Speed | BatteryV | BatteryA | ...
    """
    df = df.copy()
    df = parse_time_column(df)

    df["Value"] = pd.to_numeric(df["Value"], errors="coerce")
    wide = df.pivot_table(
        index="Time",
        columns="Parameter",
        values="Value",
        aggfunc="last"
    ).reset_index()

    if prefix:
        rename = {}
        for c in wide.columns:
            if c != "Time":
                rename[c] = f"{prefix}_{c}"
        wide = wide.rename(columns=rename)

    return wide

def process_cells(df):
    """
    Аналіз напруги комірок батареї.
    """
    df = df.copy()
    df = parse_time_column(df)

    for col in ["Module", "Volts", "InterpolatedTemperature", "SoC", "Capacity1",
                "SoCsum", "Capacity2"]:
        if col in df.columns:
            df[col] = safe_numeric(df[col])

    summary = df.groupby("Time").agg(

```

```

    cell_voltage_min=("Volts", "min"),
    cell_voltage_max=("Volts", "max"),
    cell_voltage_avg=("Volts", "mean"),
    cell_voltage_delta=("Volts", lambda x: x.max() - x.min()),
    cell_soc_avg=("SoC", "mean"),
    cell_temp_avg=("InterpolatedTemperature", "mean"),
    cell_temp_min=("InterpolatedTemperature", "min"),
    cell_temp_max=("InterpolatedTemperature", "max")
).reset_index()

return df, summary

def process_cell_temperatures(df):
    """
    Аналіз температурних датчиків батарейних модулів.
    """
    df = df.copy()
    df = parse_time_column(df)

    for col in ["Module", "Sensor", "Temperature"]:
        if col in df.columns:
            df[col] = safe_numeric(df[col])

    summary = df.groupby("Time").agg(
        module_temp_min=("Temperature", "min"),
        module_temp_max=("Temperature", "max"),
        module_temp_avg=("Temperature", "mean"),
        module_temp_delta=("Temperature", lambda x: x.max() - x.min())
    ).reset_index()

    return df, summary

# -----
# 6. Обробка інформаційного LOG-файлу
# -----

def process_info_log(path):
    """
    Витяг корисних службових повідомлень з лог-файлу.
    """
    text = read_text_file(path)
    rows = []

    for line in text.splitlines():
        match = re.match(r"^\d{2}-\d{2}-\d{4}\s+\d{2}:\d{2}:\d{2}\.\d{3}\s+(.*)$", line.strip())
        if match:

```

```

        t = pd.to_datetime(match.group(1), format="%d-%m-%Y %H:%M:%S.%f",
errors="coerce")
        message = match.group(2)
        rows.append({
            "Time": t,
            "Message": message
        })

log_df = pd.DataFrame(rows)

important = pd.DataFrame()
if not log_df.empty:
    mask = log_df["Message"].str.contains(
        "Connected|STOPPED|started|ELM327|ISO|ERROR|OK|Reset|Please|OBD|app",
        case=False,
        na=False
    )
    important = log_df[mask].copy()

return log_df, important

def merge_nearest(base, other, tolerance_seconds=3):
    """
    Об'єднання таблиць за найближчим часом.
    """
    if base is None or base.empty:
        return other

    if other is None or other.empty:
        return base

    base = base.sort_values("Time")
    other = other.sort_values("Time")

    merged = pd.merge_asof(
        base,
        other,
        on="Time",
        direction="nearest",
        tolerance=pd.Timedelta(seconds=tolerance_seconds)
    )

    return merged

def calculate_energy_metrics(df):
    """

```

```

Розрахунок ключових показників енергоспоживання електромобіля.
"""
df = df.copy()
df = df.sort_values("Time")

for col in df.columns:
    if col != "Time":
        df[col] = pd.to_numeric(df[col], errors="ignore")

if "BatteryV" in df.columns and "BatteryA" in df.columns:
    df["battery_power_kw"] = (pd.to_numeric(df["BatteryV"], errors="coerce") *
                             pd.to_numeric(df["BatteryA"], errors="coerce")) /
1000

    df["discharge_power_kw"] = df["battery_power_kw"].apply(lambda x: abs(x) if
x < 0 else 0)
    df["regen_power_kw"] = df["battery_power_kw"].apply(lambda x: x if x > 0
else 0)

if "Time" in df.columns:
    df["dt_hours"] = df["Time"].diff().dt.total_seconds() / 3600
    df["dt_hours"] = df["dt_hours"].fillna(0)
    df.loc[df["dt_hours"] < 0, "dt_hours"] = 0
    df.loc[df["dt_hours"] > 1, "dt_hours"] = 0

if "discharge_power_kw" in df.columns:
    df["energy_used_kwh_step"] = df["discharge_power_kw"] * df["dt_hours"]
    df["energy_used_kwh_total"] = df["energy_used_kwh_step"].cumsum()

if "regen_power_kw" in df.columns:
    df["energy_regen_kwh_step"] = df["regen_power_kw"] * df["dt_hours"]
    df["energy_regen_kwh_total"] = df["energy_regen_kwh_step"].cumsum()

if "Speed" in df.columns and "dt_hours" in df.columns:
    df["distance_km_step"] = pd.to_numeric(df["Speed"],
errors="coerce").fillna(0) * df["dt_hours"]
    df["distance_km_total"] = df["distance_km_step"].cumsum()

if "energy_used_kwh_total" in df.columns and "distance_km_total" in df.columns:
    df["consumption_kwh_100km"] = np.where(
        df["distance_km_total"] > 0,
        df["energy_used_kwh_total"] / df["distance_km_total"] * 100,
        np.nan
    )

if "Speed" in df.columns:
    speed = pd.to_numeric(df["Speed"], errors="coerce").fillna(0)

conditions = [
    speed == 0,

```

```

        (speed > 0) & (speed <= 30),
        (speed > 30) & (speed <= 70),
        (speed > 70) & (speed <= 110),
        speed > 110
    ]

    labels = [
        "Стоянка / холостий режим",
        "Міський повільний рух",
        "Міський або приміський режим",
        "Трасовий режим",
        "Високошвидкісний режим"
    ]

    df["drive_mode"] = np.select(conditions, labels, default="Невизначений режим")

    return df

def detect_anomalies(df):
    """
    Виявлення потенційно проблемних або важливих режимів.
    """
    anomalies = []

    if "BatteryV" in df.columns:
        bad = df[(pd.to_numeric(df["BatteryV"], errors="coerce") < 250) |
                 (pd.to_numeric(df["BatteryV"], errors="coerce") > 450)]
        for _, r in bad.iterrows():
            anomalies.append([r["Time"], "Аномальна напруга батареї",
                             r.get("BatteryV")])

    if "BatteryA" in df.columns:
        current = pd.to_numeric(df["BatteryA"], errors="coerce")
        bad = df[current.abs() > 200]
        for _, r in bad.iterrows():
            anomalies.append([r["Time"], "Високий струм батареї",
                             r.get("BatteryA")])

    if "BatteryT" in df.columns:
        temp = pd.to_numeric(df["BatteryT"], errors="coerce")
        bad = df[(temp < -20) | (temp > 55)]
        for _, r in bad.iterrows():
            anomalies.append([r["Time"], "Небажана температура батареї",
                             r.get("BatteryT")])

    if "cell_voltage_delta" in df.columns:
        delta = pd.to_numeric(df["cell_voltage_delta"], errors="coerce")

```

```

        bad = df[delta > 0.08]
        for _, r in bad.iterrows():
            anomalies.append([r["Time"], "Підвищений розбаланс комірок батареї",
r.get("cell_voltage_delta")])

    if "module_temp_delta" in df.columns:
        delta = pd.to_numeric(df["module_temp_delta"], errors="coerce")
        bad = df[delta > 8]
        for _, r in bad.iterrows():
            anomalies.append([r["Time"], "Нерівномірність температури модулів",
r.get("module_temp_delta")])

    return pd.DataFrame(anomalies, columns=["Time", "Anomaly", "Value"])

def build_summary(df):
    """
    Формування аналітичного звіту.
    """
    summary = {}

    summary["Кількість записів"] = len(df)

    if "Time" in df.columns and df["Time"].notna().any():
        summary["Початок запису"] = df["Time"].min()
        summary["Кінець запису"] = df["Time"].max()
        summary["Тривалість запису, хв"] = round((df["Time"].max() -
df["Time"].min()).total_seconds() / 60, 2)

    if "Speed" in df.columns:
        speed = pd.to_numeric(df["Speed"], errors="coerce")
        summary["Середня швидкість, км/год"] = round(speed.mean(), 2)
        summary["Максимальна швидкість, км/год"] = round(speed.max(), 2)

    if "BatteryV" in df.columns:
        v = pd.to_numeric(df["BatteryV"], errors="coerce")
        summary["Середня напруга батареї, В"] = round(v.mean(), 2)
        summary["Мінімальна напруга батареї, В"] = round(v.min(), 2)
        summary["Максимальна напруга батареї, В"] = round(v.max(), 2)

    if "BatteryA" in df.columns:
        a = pd.to_numeric(df["BatteryA"], errors="coerce")
        summary["Середній струм батареї, А"] = round(a.mean(), 2)
        summary["Максимальний струм розряду, А"] = round(abs(a.min()), 2)
        summary["Максимальний струм рекуперації, А"] = round(a.max(), 2)

    if "battery_power_kw" in df.columns:
        p = pd.to_numeric(df["battery_power_kw"], errors="coerce")
        summary["Середня потужність батареї, кВт"] = round(p.mean(), 3)

```

```

summary["Максимальна потужність розряду, кВт"] = round(abs(p.min()), 3)
summary["Максимальна потужність рекуперації, кВт"] = round(p.max(), 3)

if "energy_used_kwh_total" in df.columns:
    summary["Витрачена енергія, кВт·год"] =
round(df["energy_used_kwh_total"].max(), 4)

if "energy_regen_kwh_total" in df.columns:
    summary["Рекуперована енергія, кВт·год"] =
round(df["energy_regen_kwh_total"].max(), 4)

if "distance_km_total" in df.columns:
    summary["Розрахована дистанція, км"] = round(df["distance_km_total"].max(),
4)

if "consumption_kwh_100km" in df.columns:
    valid = df["consumption_kwh_100km"].dropna()
    if len(valid) > 0:
        summary["Середнє споживання, кВт·год/100 км"] = round(valid.iloc[-1],
3)

if "SoC1" in df.columns:
    soc = pd.to_numeric(df["SoC1"], errors="coerce")
    summary["SOC початковий, %"] = round(soc.dropna().iloc[0], 2) if
len(soc.dropna()) else np.nan
    summary["SOC кінцевий, %"] = round(soc.dropna().iloc[-1], 2) if
len(soc.dropna()) else np.nan
    summary["Зміна SOC, %"] = round(soc.dropna().iloc[0] - soc.dropna().iloc[-
1], 2) if len(soc.dropna()) else np.nan

if "BatteryT" in df.columns:
    t = pd.to_numeric(df["BatteryT"], errors="coerce")
    summary["Середня температура батареї, °C"] = round(t.mean(), 2)
    summary["Мінімальна температура батареї, °C"] = round(t.min(), 2)
    summary["Максимальна температура батареї, °C"] = round(t.max(), 2)

if "cell_voltage_delta" in df.columns:
    summary["Максимальний розбаланс комірок, В"] =
round(df["cell_voltage_delta"].max(), 4)

return pd.DataFrame(list(summary.items()), columns=["Показник", "Значення"])

def plot_results(df, output_dir="ev_output"):
    os.makedirs(output_dir, exist_ok=True)

    plots = []

    def save_plot(filename, title):

```

```

path = os.path.join(output_dir, filename)
plt.title(title)
plt.grid(True)
plt.tight_layout()
plt.savefig(path, dpi=160)
plt.show()
plots.append(path)

if "Time" in df.columns and "Speed" in df.columns:
    plt.figure(figsize=(12, 5))
    plt.plot(df["Time"], pd.to_numeric(df["Speed"], errors="coerce"))
    plt.xlabel("Час")
    plt.ylabel("Швидкість, км/год")
    save_plot("01_speed.png", "Графік зміни швидкості електромобіля")

if "Time" in df.columns and "battery_power_kw" in df.columns:
    plt.figure(figsize=(12, 5))
    plt.plot(df["Time"], pd.to_numeric(df["battery_power_kw"],
errors="coerce"))
    plt.xlabel("Час")
    plt.ylabel("Потужність, кВт")
    save_plot("02_power.png", "Графік миттєвої потужності батареї")

if "Time" in df.columns and "SoC1" in df.columns:
    plt.figure(figsize=(12, 5))
    plt.plot(df["Time"], pd.to_numeric(df["SoC1"], errors="coerce"))
    plt.xlabel("Час")
    plt.ylabel("SOC, %")
    save_plot("03_soc.png", "Графік зміни рівня заряду батареї")

if "Time" in df.columns and "BatteryT" in df.columns:
    plt.figure(figsize=(12, 5))
    plt.plot(df["Time"], pd.to_numeric(df["BatteryT"], errors="coerce"))
    plt.xlabel("Час")
    plt.ylabel("Температура, °C")
    save_plot("04_battery_temperature.png", "Графік температури батареї")

if "distance_km_total" in df.columns and "consumption_kwh_100km" in df.columns:
    valid = df.dropna(subset=["consumption_kwh_100km"])
    if len(valid) > 0:
        plt.figure(figsize=(12, 5))
        plt.plot(valid["distance_km_total"], valid["consumption_kwh_100km"])
        plt.xlabel("Дистанція, км")
        plt.ylabel("кВт·год/100 км")
        save_plot("05_consumption.png", "Графік питомого енергоспоживання")

if "drive_mode" in df.columns and "energy_used_kwh_step" in df.columns:
    mode_energy =
df.groupby("drive_mode")["energy_used_kwh_step"].sum().sort_values(ascending=False)
    if len(mode_energy) > 0:

```

```

plt.figure(figsize=(10, 5))
mode_energy.plot(kind="bar")
plt.xlabel("Режим руху")
plt.ylabel("Витрачена енергія, кВт·год")
save_plot("06_energy_by_mode.png", "Розподіл енергоспоживання за
режимами руху")

if "Time" in df.columns and "cell_voltage_delta" in df.columns:
plt.figure(figsize=(12, 5))
plt.plot(df["Time"], pd.to_numeric(df["cell_voltage_delta"],
errors="coerce"))
plt.xlabel("Час")
plt.ylabel("ΔU, В")
save_plot("07_cell_balance.png", "Графік розбалансу напруги комірок
батареї")

return plots

def save_outputs(output_dir, tables):
"""
Збереження всіх результатів у Excel та ZIP.
"""
os.makedirs(output_dir, exist_ok=True)

excel_path = os.path.join(output_dir, "decoded_ev_analysis.xlsx")

with pd.ExcelWriter(excel_path, engine="openpyxl") as writer:
for name, table in tables.items():
if table is not None and not table.empty:
safe_name = re.sub(r"^[A-Za-zA-яa-я0-9_]", "_", name)[:31]
table.to_excel(writer, sheet_name=safe_name, index=False)

zip_path = "ev_decoded_results.zip"

with zipfile.ZipFile(zip_path, "w", zipfile.ZIP_DEFLATED) as zipf:
for root, dirs, files in os.walk(output_dir):
for file in files:
full = os.path.join(root, file)
arc = os.path.relpath(full, output_dir)
zipf.write(full, arc)

return excel_path, zip_path

def main():
print("=====")
print(" СИСТЕМА РОЗШИФРУВАННЯ ТХТ-ДАНИХ ЕЛЕКТРОМОБІЛЯ")
print("=====")
print("Завантажте всі ТХТ-файли одним набором.")
print("Програма автоматично визначить тип кожного файлу.")
print("")

```

```

paths = upload_files()

if not paths:
    print("Файли не знайдено.")
    return

output_dir = "ev_output"
os.makedirs(output_dir, exist_ok=True)

loaded_tables = {}
decoded_pid_tables = []
obd_wide = pd.DataFrame()
calc_wide = pd.DataFrame()
cells_raw = pd.DataFrame()
cells_summary = pd.DataFrame()
temp_raw = pd.DataFrame()
temp_summary = pd.DataFrame()
logs = pd.DataFrame()
important_logs = pd.DataFrame()

print("\nВиявлені файли:")
for p in paths:
    text = read_text_file(p)
    ftype = detect_file_type(p, text)
    print(f" - {p} --> {ftype}")

    try:
        if ftype in ["pid_hex", "pid_int"]:
            df = read_semicolon_table(p)
            decoded = decode_pid_table(df, ftype)
            decoded["source_file"] = os.path.basename(p)
            decoded_pid_tables.append(decoded)
            loaded_tables[f"decoded_{ftype}"] = decoded

        elif ftype == "obd":
            df = read_semicolon_table(p)
            obd_wide = pivot_parameter_file(df, prefix="")
            loaded_tables["obd_raw"] = df
            loaded_tables["obd_decoded"] = obd_wide

        elif ftype == "calc":
            df = read_semicolon_table(p)
            calc_wide = pivot_parameter_file(df, prefix="calc")
            loaded_tables["calc_raw"] = df
            loaded_tables["calc_decoded"] = calc_wide

        elif ftype == "parameter":
            df = read_semicolon_table(p)
            param_wide = pivot_parameter_file(df, prefix="param")

```

```

loaded_tables[f"parameter_{os.path.basename(p)[:15]}"] = param_wide

elif ftype == "cells":
    df = read_semicolon_table(p)
    cells_raw, cells_summary = process_cells(df)
    loaded_tables["cells_raw"] = cells_raw
    loaded_tables["cells_summary"] = cells_summary

elif ftype == "cell_temperatures":
    df = read_semicolon_table(p)
    temp_raw, temp_summary = process_cell_temperatures(df)
    loaded_tables["cell_temperatures_raw"] = temp_raw
    loaded_tables["cell_temperatures_summary"] = temp_summary

elif ftype == "info_log":
    logs, important_logs = process_info_log(p)
    loaded_tables["info_log"] = logs
    loaded_tables["important_log"] = important_logs

else:
    print(f"    Увага: тип файлу не розпізнано: {p}")

except Exception as e:
    print(f"    Помилка обробки файлу {p}: {e}")

if decoded_pid_tables:
    decoded_pid_all = pd.concat(decoded_pid_tables, ignore_index=True)
    loaded_tables["pid_all_decoded"] = decoded_pid_all
else:
    decoded_pid_all = pd.DataFrame()

print("\nФормування загальної таблиці аналізу...")

main_df = pd.DataFrame()

if not obd_wide.empty:
    main_df = obd_wide.copy()

if not calc_wide.empty:
    main_df = merge_nearest(main_df, calc_wide)

if not cells_summary.empty:
    main_df = merge_nearest(main_df, cells_summary)

if not temp_summary.empty:
    main_df = merge_nearest(main_df, temp_summary)

if main_df.empty:
    print("Не вдалося сформувати загальну таблицю. Перевірте наявність OBD або
CALC файлів.")

```

```

    return

    main_df = calculate_energy_metrics(main_df)
    anomalies = detect_anomalies(main_df)
    summary = build_summary(main_df)

    loaded_tables["main_decoded_dataset"] = main_df
    loaded_tables["summary"] = summary
    loaded_tables["anomalies"] = anomalies

    print("\n=====")
    print(" ПІДСУМКОВИЙ АНАЛІТИЧНИЙ ЗВІТ")
    print("=====")
    print(summary.to_string(index=False))

    if not anomalies.empty:
        print("\nВиявлені потенційні аномалії:")
        print(anomalies.head(30).to_string(index=False))
    else:
        print("\nКритичних аномалій за встановленими правилами не виявлено.")

    print("\nПобудова графіків...")
    plot_results(main_df, output_dir=output_dir)

    excel_path, zip_path = save_outputs(output_dir, loaded_tables)

    print("\n=====")
    print(" ОБРОБКУ ЗАВЕРШЕНО")
    print("=====")
    print(f"Excel-файл з розшифрованими таблицями: {excel_path}")
    print(f"ZIP-архів з результатами: {zip_path}")

    try:
        from google.colab import files
        files.download(zip_path)
    except Exception:
        print("Автоматичне завантаження ZIP недоступне. Файл збережено у папці запуску.")

main()

```